

需求管理

API 参考

文档版本 26
发布日期 2024-08-30



版权所有 © 华为云计算技术有限公司 2024。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为云计算技术有限公司

地址：贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址：<https://www.huaweicloud.com/>

目录

1 使用前必读	1
1.1 概述	1
1.2 调用说明	1
1.3 终端节点	1
1.4 基本概念	2
2 API 概览	4
3 如何调用 API	5
3.1 构造请求	5
3.2 认证鉴权	8
3.3 返回结果	9
4 API	11
4.1 Scrum 项目的工作项	11
4.1.1 创建工作项类型自定义字段	11
4.1.2 上传图片	17
4.1.3 获取工作项完成率	22
4.1.4 获取指定工作项的评论列表	26
4.1.5 按用户查询工时（单项目）	29
4.1.6 按用户查询工时（多项目）	35
4.1.7 添加指定工作项工时	41
4.1.8 查询项目下的工时类型	48
4.1.9 获取工作项历史记录	51
4.1.10 创建工作项	55
4.1.11 查询项目的工作项	68
4.1.12 高级查询工作项	76
4.1.13 批量删除工作项	91
4.1.14 查询工作项详情	95
4.1.15 更新工作项	104
4.1.16 删除工作项	115
4.1.17 获取子工作项	117
4.1.18 查询项目下所有工作项的历史记录	124
4.1.19 查询 Scrum 项目的工作项流转配置	128
4.1.20 细粒度权限用户创建工作项	131

4.1.21 查询当前工作项已经关联的工作项.....	142
4.1.22 查询当前工作项已经关联的关联 Wiki.....	148
4.1.23 查询当前工作项已经关联的代码提交记录 / 分支创建记录.....	153
4.1.24 查询关联用例.....	157
4.1.25 查询 Scrum 工作项自定义字段.....	162
4.1.26 下载图片.....	168
4.1.27 上传工作项附件.....	171
4.1.28 下载工作项附件.....	176
4.1.29 删除附件.....	180
4.1.30 查询迭代下工作项状态的统计数据（处理人维度）.....	181
4.1.31 获取指定工作项停留时间.....	187
4.1.32 高级查询我的待办工作项.....	192
4.2 Scrum 项目的迭代.....	200
4.2.1 获取指定项目的迭代列表.....	200
4.2.2 批量删除项目的迭代.....	203
4.2.3 查看迭代详情.....	208
4.2.4 创建 Scrum 项目迭代.....	211
4.2.5 更新 Scrum 项目迭代.....	216
4.2.6 删除项目迭代.....	221
4.2.7 查看迭代历史记录.....	222
4.3 用户信息.....	228
4.3.1 获取当前用户信息.....	228
4.3.2 更新用户昵称.....	230
4.3.3 获取当前用户角色.....	234
4.3.4 更新子用户昵称.....	238
4.4 项目成员.....	242
4.4.1 主动退出项目.....	242
4.4.2 获取指定项目的成员用户列表.....	244
4.4.3 批量添加项目成员.....	247
4.4.4 批量删除项目成员.....	252
4.4.5 更新成员在项目中的角色.....	256
4.4.6 添加项目成员.....	260
4.4.7 AGC 调用 当前用户申请加入项目.....	265
4.5 项目信息.....	267
4.5.1 获取项目详情.....	267
4.5.2 更新项目.....	272
4.5.3 删除项目.....	276
4.5.4 查询项目列表.....	279
4.5.5 获取租户没有加入的项目.....	285
4.5.6 创建项目.....	291
4.5.7 检查项目名称是否存在.....	296
4.5.8 查询项目模板.....	300

4.6 项目指标.....	305
4.6.1 查询人均 bug.....	305
4.6.2 查询需求按时完成率.....	312
4.6.3 查询缺陷密度.....	318
4.7 项目统计.....	324
4.7.1 获取 bug 统计信息.....	324
4.7.2 获取需求统计信息.....	329
4.7.3 获取项目概览.....	334
4.8 Scrum 项目的模块.....	338
4.8.1 创建项目的模块.....	338
4.8.2 查询项目的模块列表.....	343
4.8.3 更新项目的模块.....	349
4.8.4 删除项目的模块.....	354
4.9 Scrum 项目的领域.....	356
4.9.1 创建项目的领域.....	356
4.9.2 查询项目的领域列表.....	361
4.9.3 更新项目的领域.....	365
4.9.4 取消领域与项目的关联关系.....	370
4.10 看板项目的工作项.....	371
4.10.1 查询看板项目下工作项的状态历史记录.....	372
4.10.2 查询看板项目下的工作项.....	376
4.10.3 查询看板项目的工作项流转配置.....	382
4.11 Scrum 项目的状态.....	386
4.11.1 查询项目的状态列表.....	386
5 历史 API (待下线)	392
5.1 项目成员角色待下线接口.....	392
6 应用示例.....	398
6.1 示例 1: 获取需求管理中项目列表.....	398
6.2 示例 2: 查询指定项目下的迭代列表.....	399
6.3 示例 3: 查询用户工作项的评论.....	400
7 附录.....	403
7.1 状态码.....	403
7.2 错误码.....	405
7.3 获取项目 ID.....	410
7.4 获取账号 ID.....	411
8 文档修订记录.....	413

1 使用前必读

1.1 概述

需求管理（CodeArts Req）为敏捷开发团队提供简单高效的团队协作服务，包含多项目的管理、敏捷迭代、看板协作、需求管理、缺陷跟踪、文档管理、Wiki在线协作、仪表盘自定义报表等功能。

您可以使用本文档提供API对需求管理进行相关操作，如创建、删除、查询任务等，支持的全部操作请参见[API概览](#)。

在调用API之前，请确保已经充分了解需求管理[基本概念](#)，详细信息请参见[产品介绍](#)。

1.2 调用说明

需求管理提供了REST（Representational State Transfer）风格API，支持您通过HTTPS请求调用，调用方法请参见[如何调用API](#)。

1.3 终端节点

终端节点即调用API的[请求地址](#)，不同服务不同区域的终端节点不同。

需求管理的终端节点如[表1-1](#)所示，请您根据业务需要选择对应区域的终端节点。

表 1-1 需求管理的终端节点

区域名称	区域	终端节点（Endpoint）
华北-北京一	cn-north-1	projectman-ext.cn-north-1.myhuaweicloud.com
华北-北京四	cn-north-4	projectman-ext.cn-north-4.myhuaweicloud.com
华东-上海一	cn-east-3	projectman-ext.cn-east-3.myhuaweicloud.com

区域名称	区域	终端节点 (Endpoint)
华东-上海二	cn-east-2	projectman-ext.cn-east-2.myhuaweicloud.com
华南-广州	cn-south-1	projectman-ext.cn-south-1.myhuaweicloud.com
西南-贵阳一	cn-southwest-2	projectman-ext.cn-southwest-2.myhuaweicloud.com

1.4 基本概念

- 账号
用户注册华为云时的账号，账号对其所拥有的资源及云服务具有完全的访问权限，可以重置用户密码、分配用户权限等。由于账号是付费主体，为了确保账号安全，建议您不要直接使用账号进行日常管理工作，而是创建用户并使用他们进行日常管理工作。
- 用户
由账号在IAM中创建的用户，是云服务的使用人员，具有身份凭证（密码和访问密钥）。
在[我的凭证](#)下，您可以查看账号ID和用户ID。通常在调用API的鉴权过程中，您需要用到账号、用户和密码等信息。
- 区域 (Region)
从地理位置和网络时延维度划分，同一个Region内共享弹性计算、块存储、对象存储、VPC网络、弹性公网IP、镜像等公共服务。Region分为通用Region和专属Region。
 - 通用Region指面向公共租户提供通用云服务的Region。
 - 专属Region指只承载同一类业务或只面向特定租户提供业务服务的专用Region。
- 可用区 (AZ, Availability Zone)
一个AZ是一个或多个物理数据中心的集合，有独立的风火水电，AZ内逻辑上再将计算、网络、存储等资源划分成多个集群。一个Region中的多个AZ间通过高速光纤相连，以满足用户跨AZ构建高可用性系统的需求。
- 项目
用户在需求管理中创建的项目，根据模板创建项目，根据您选择的项目模板，还包含有预置的工作项、代码或资源等。通过管理项目IAM用户（即“团队成员”）所需的权限。将IAM用户作为团队成员添加到项目，给每个团队成员分配角色并赋予项目及其资源的操作权限。
- 项目模板
使用模板创建项目，模板项目是指**默认预置模板类型**的项目，即基于Scrum和看板形式，由需求管理预置好一些工作项和流程。选择项目模板后，会自动生成对应样例模板，供用户参考和使用，用户也可以新建自己的开发任务。
- 企业项目
企业项目是项目的升级版，针对企业不同项目间资源的分组和管理，是逻辑隔离。企业项目中可以包含多个区域的资源，且项目中的资源可以迁入迁出。

关于企业项目ID的获取及企业项目特性的详细信息，请参见《[企业管理服务用户指南](#)》。

2 API 概览

表 2-1 API 概览

类型	说明
Scrum项目的工作项	Scrum项目工作项的创建、删除、修改、查询等相关接口。
Scrum项目的迭代	Scrum项目迭代的创建、删除、修改、查询等相关接口。
用户信息	获取当前用户信息，更新用户昵称，获取当前用户角色。
项目成员	添加、查询、删除、退出项目成员等接口信息。
项目信息	创建、更新、查询、删除项目等相关的接口。
项目指标	查询人均bug，查询需求按时完成率，查询缺陷密度。
项目统计	获取bug，需求统计信息，获取项目概览。
看板项目的工作项	查询看板项目下工作项的相关接口。
Scrum项目的模块	Scrum项目模块的创建、查询、更新和删除相关接口。
Scrum项目的领域	Scrum项目领域的创建、查询、更新和删除相关接口。

3 如何调用 API

3.1 构造请求

本节介绍REST API请求的组成，并以调用IAM服务的**获取用户Token**获取请求认证接口说明如何调用API，该API获取用户的Token，Token可以用于调用其他API时鉴权。

您还可以通过这个视频教程了解如何构造请求调用API：<https://bbs.huaweicloud.com/videos/102987>。

请求 URI

尽管请求URI包含在请求消息头中，但大多数语言或框架都要求您从请求消息中单独传递它，所以在此单独强调。

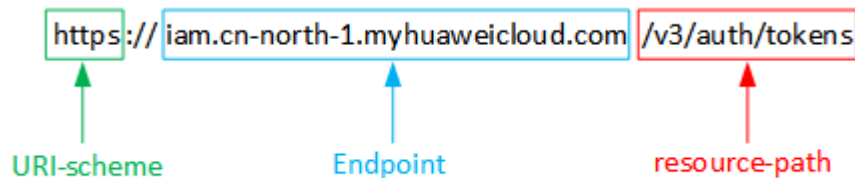
请求URI由四部分构成：**{URI-scheme}://{Endpoint} / {resource-path} ? {query-string}**

参数	说明
URI-scheme	表示用于传输请求的协议，当前所有API均采用HTTPS协议。
Endpoint	指定承载REST服务端点的服务器域名或IP，不同服务不同区域的Endpoint不同，您可以从 终端节点 中获取。 例如IAM服务在“华北-北京一”区域的Endpoint为“iam.cn-north-1.myhuaweicloud.com”。
resource-path	资源路径，即API访问路径，从具体API的URI模块获取。例如，获取用户Token接口的resource-path为 /v3/auth/tokens 。
query-string	查询参数，是可选部分，并不是每个API都有查询参数。查询参数前面需要带一个？，形式为 参数名=参数取值 。例如， limit=10 表示查询不超过10条数据。

例如，您需要获取IAM在“华北-北京一”区域的Token，则需使用“华北-北京一”区域的Endpoint（iam.cn-north-1.myhuaweicloud.com），并在**获取用户Token**获取请求认证的URI部分找到resource-path（/v3/auth/tokens），拼接起来如下所示。

```
https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens
```

图 3-1 URI 示意图



说明

为查看方便，在每个具体API的URI部分，只给出resource-path部分，并将请求方法写在一起。这是因为URI-scheme都是HTTPS，而Endpoint在同一个区域也相同，所以简洁起见将这两部分省略。

请求方法

HTTP请求方法（也称为操作或动词），它告诉服务你正在请求什么类型的操作。

- **GET**：请求服务器返回指定资源。
- **PUT**：请求服务器更新指定资源。
- **POST**：请求服务器新增资源或执行特殊操作。
- **DELETE**：请求服务器删除指定资源，如删除对象等。
- **HEAD**：请求服务器资源头部。
- **PATCH**：请求服务器更新资源的部分内容。当资源不存在的时候，PATCH可能会去创建一个新的资源。

在**获取用户Token**获取请求认证的URI部分，您可以看到其请求方法为“POST”，则其请求为：

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens
```

请求消息头

附加请求头字段，如指定的URI和HTTP方法所要求的字段。例如定义消息体类型的请求头“Content-Type”，请求鉴权信息等。

如下公共消息头需要添加到请求中。

- **Content-Type**：消息体的类型（格式），必选，默认取值为“application/json”，有其他取值时会在具体接口中专门说明。
- **X-Auth-Token**：用户Token，可选，当使用Token方式认证时，必须填充该字段。用户Token也就是调用**获取用户Token**获取请求认证接口的响应值，该接口是唯一不需要认证的接口。

说明

公有云API同时支持使用AK/SK认证，AK/SK认证是使用SDK对请求进行签名，签名过程会自动往请求中添加Authorization（签名认证信息）和X-Sdk-Date（请求发送的时间）请求头。

AK/SK认证的详细说明请参见[认证鉴权](#)。

- **X-Project-ID**: 子项目ID, 可选, 在多项目场景中使用。
- **X-Domain-ID**: 账号ID。

对于**获取用户Token**获取请求认证接口, 由于不需要认证, 所以只添加“Content-Type”即可, 添加消息头后的请求如下所示。

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

请求消息体

请求消息体通常以结构化格式发出, 与请求消息头中Content-type对应, 传递除请求消息头之外的内容。

每个接口的请求消息体内容不同, 也并不是每个接口都需要有请求消息体 (或者说消息体为空), GET、DELETE操作类型的接口就不需要消息体, 消息体具体内容需要根据具体接口而定。

对于**获取用户Token**获取请求认证接口, 您可以从接口的请求部分看到所需的请求参数及参数说明。将消息体加入后的请求如下所示, 加粗的斜体字段需要根据实际值填写, 其中***username***为用户名, ***domainname***为用户所属的账号名称, ***********为用户登录密码, ***xxxxxxxx***为project的名称, 如cn-north-1, 您可以从**终端节点**中获取。

📖 说明

scope参数定义了Token的作用域, 下面示例中获取的Token仅能访问project下的资源。您还可以设置Token作用域为某个账号下所有资源或账号的某个project下的资源, 详细定义请参见**获取用户Token**获取请求认证。

```
POST https://iam.cn-north-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "*****",
          "domain": {
            "name": "domainname"
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "xxxxxxxx"
      }
    }
  }
}
```

到这里为止这个请求需要的内容就具备齐全了, 您可以使用**curl**、**Postman**或直接编写代码等方式发送请求调用API。对于**获取用户Token**获取请求认证接口, 返回的响应消息头中“x-subject-token”就是需要获取的用户Token。有了Token之后, 您就可以使用Token认证调用其他API。

3.2 认证鉴权

调用接口有如下两种认证方式，您可以选择其中一种进行认证鉴权。

- Token认证：通过Token认证通用请求。
- AK/SK认证：通过AK（Access Key ID）/SK（Secret Access Key）加密调用请求。

Token 认证

📖 说明

Token的有效期为24小时，需要使用一个Token鉴权时，可以先缓存起来，避免频繁调用。

Token在计算机系统中代表令牌（临时）的意思，拥有Token就代表拥有某种权限。Token认证就是在调用API的时候将Token加到请求消息头，从而通过身份认证，获得操作API的权限。

Token可通过调用[获取用户Token](#)接口获取，调用本服务API需要project级别的Token，即调用接口时，请求body中auth.scope的取值需要选择project，如下所示。

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "*****#",
          "domain": {
            "name": "domainname"
          }
        }
      }
    }
  },
  "scope": {
    "project": {
      "name": "xxxxxxx"
    }
  }
}
```

获取Token后，再调用其他接口时，您需要在请求消息头中添加“X-Auth-Token”，其值即为Token。例如Token值为“ABCDEFJ...”，则调用接口时将“X-Auth-Token: ABCDEFJ...”加到请求消息头即可，如下所示。

```
GET https://iam.cn-north-1.myhuaweicloud.com/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

您还可以通过这个视频教程了解如何使用Token认证：<https://bbs.huaweicloud.com/videos/101333>。

AK/SK 认证

📖 说明

AK/SK签名认证方式仅支持消息体大小12M以内，12M以上的请求请使用Token认证。

AK/SK认证就是使用AK/SK对请求进行签名，在请求时将签名信息添加到消息头，从而通过身份认证。

- AK(Access Key ID)：访问密钥ID。与私有访问密钥关联的唯一标识符；访问密钥ID和私有访问密钥一起使用，对请求进行加密签名。
- SK(Secret Access Key)：与访问密钥ID结合使用的密钥，对请求进行加密签名，可标识发送方，并防止请求被修改。

使用AK/SK认证时，您可以基于签名算法使用AK/SK对请求进行签名，也可以使用专门的签名SDK对请求进行签名。详细的签名方法和SDK使用方法请参见[API签名指南](#)。

须知

签名SDK只提供签名功能，与服务提供的SDK不同，使用时请注意。

3.3 返回结果

状态码

请求发送以后，您会收到响应，包含状态码、响应消息头和消息体。

状态码是一组从1xx到5xx的数字代码，状态码表示了请求响应的状态，完整的状态码列表请参见[状态码](#)。

对于[获取用户Token](#)获取请求认证接口，如果调用后返回状态码为“201”，则表示请求成功。

响应消息头

对应请求消息头，响应同样也有消息头，如“Content-type”。

对于[获取用户Token](#)获取请求认证接口，返回如[图3-2](#)所示的消息头，其中“x-subject-token”就是需要获取的用户Token。有了Token之后，您就可以使用Token认证调用其他API。

图 3-2 获取用户 Token 响应消息头

```
connection → keep-alive
content-type → application/json
date → Tue, 12 Feb 2019 06:52:13 GMT
server → Web Server
strict-transport-security → max-age=31536000; includeSubdomains;
transfer-encoding → chunked
via → proxy A
x-content-type-options → nosniff
x-download-options → noopen
x-frame-options → SAMEORIGIN
x-iam-trace-id → 218d45ab-d674-4995-af3a-2d0255ba41b5
x-subject-token → MIIVXQYJKoZIhvcNAQcCoIIYJTCCGEOCAQExDTALBglghkgBZQMEAgEwgharBgkqhkiG9w0BBwGgghacBIIWmHsidG9rZW4iOensiZXhwaXJlc19hdCI6IiwMTktMDItMTNUMC
fj3Kjs6YgKnpVNRbW2eZ5eb785Z0kajACgkqO1wi4JIGzrpd18LGXK5bdfq4lqHCYb8P4NaYONYejeAgzVefYtLWT1GSO0zxKZmlQHq82HBqHdglZO9fuEbL5dMhdavj+33wEI
xHRCE9I87o+k9-
j+CMZSEB7bUGd5Uj6eRASXl1jipPEGA270g1FruooL6jggIFkNPQuFSOU8+uSsttVwRtnfsC+qTp22Rkd5MCqFGQ8LcuUxC3a+9CMBnOintWW7oeRUUVhVpxk8pxiX1wTEboX-
RzT6MUbpvGw-oPNFYxJECKnoH3HRozv0vN--n5d6Nbxg==
x-xss-protection → 1; mode=block;
```

响应消息体

响应消息体通常以结构化格式返回，与响应消息头中Content-type对应，传递除响应消息头之外的内容。

对于[获取用户Token](#)获取请求认证接口，返回如下消息体。为篇幅起见，这里只展示部分内容。

```
{
  "token": {
    "expires_at": "2019-02-13T06:52:13.855000Z",
    "methods": [
      "password"
    ],
    "catalog": [
      {
        "endpoints": [
          {
            "region_id": "cn-north-1",
            .....

```

当接口调用出错时，会返回错误码及错误信息说明，错误响应的Body体格式如下所示。

```
{
  "error_msg": "The format of message is error",
  "error_code": "AS.0001"
}
```

其中，error_code表示错误码，error_msg表示错误描述信息。

4 API

4.1 Scrum 项目的工作项

4.1.1 创建工作项类型自定义字段

功能介绍

创建工作项类型自定义字段

调用方法

请参见[如何调用API](#)。

URI

POST /v3/{project_id}/custom-fields

表 4-1 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<p>参数解释：项目的32位uuid，项目唯一标识，可以通过ListProjectsV4接口查询项目列表</p> <p>约束限制：正则表达式，<code>^[A-Za-z0-9]{32}\$</code></p> <p>取值范围：字符串，最小长度：32，最大长度：32</p> <p>默认取值：不涉及</p>

请求参数

表 4-2 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	<p>参数解释：用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。</p> <p>约束限制：不涉及</p> <p>取值范围：最小长度：10，最大长度：32768</p> <p>默认取值：不涉及</p>

表 4-3 请求 Body 参数

参数	是否必选	参数类型	描述
name	是	String	<p>参数解释：自定义字段名称</p> <p>约束限制：正则表达式：<code>^[a-zA-Z0-9\u4E00-\u9FA5\-_]{1,15}\$</code></p> <p>取值范围：最小长度：1，最大长度：15</p> <p>默认取值：不涉及</p>
type	是	String	<p>参数解释：自定义字段类型</p> <p>约束限制：正则表达式，<code>^(textArea select radio text checkbox date time_date number)\$</code></p> <p>对应参数解释：textArea 多行文本，select 下拉框，radio 单选框，text 单行文本，checkbox 多选框，date 日期，time_date 日期（包含时分，包含时分的日期类型的字段显示格式为：2019/10/10 10:10），number 数字</p> <p>取值范围：最小长度：4，最大长度：9</p> <p>默认取值：不涉及</p>

参数	是否必选	参数类型	描述
options	否	String	参数解释: 自定义字段类型 约束限制: 正则表达式, ^[\s\S]{1,6000}\$ 取值范围: 最小长度: 1, 最大长度: 6000 默认取值: 不涉及
memo	否	String	参数解释: 自定义字段描述 约束限制: 不涉及 取值范围: 最小长度: 0, 最大长度: 255 默认取值: 不涉及
scrum_type	否	String	参数解释: 当前新建的自定义字段应用的工作项类型 约束限制: 正则表达式, ^(Epic Feature Story Task Bug)\$ 取值范围: 最小长度: 0, 最大长度: 7 默认取值: 不涉及

响应参数

状态码: 200

表 4-4 响应 Body 参数

参数	参数类型	描述
options	String	字段可选值
region	String	区域
id	Integer	自定义字段ID
identifier	String	自定义字段唯一标识ID, 可通过 ListIssueCustomFields接口查询Scrum工作项
project_id	Integer	项目的32位uuid, 项目唯一标识, 可以通过 ListProjectsV4接口查询项目列表
tracker_id	Integer	工作项类型id 2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story
custom_field	String	自定义字编号
type	String	自定义字段类型
name	String	自定义字段名称

参数	参数类型	描述
sort	Integer	显示排序
memo	String	自定义字段描述
created	String	创建时间
modified	String	修改时间
is_delete	Boolean	是否被删除

状态码： 400

表 4-5 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-6 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Post <https://{endpoint}/v3/0792b7d26cb74bd799cbae80063dc74a/custom-fields>

```
{
  "name" : "test",
  "type" : "text",
  "memo" : "描述",
  "options" : "0792b7d26cb74bd799cbae80063dc74a",
  "scrum_type" : "Story"
}
```

响应示例

状态码： 200

OK

```
{
  "options" : "select1,select2",
  "region" : "southchina",
}
```

```
"id" : 1,  
"identifier" : "81abca040032407bb62d52536d69ce2d",  
"project_id" : 2282492,  
"tracker_id" : 3,  
"custom_field" : "custom_field34",  
"type" : "select",  
"name" : "test",  
"sort" : 1,  
"memo" : "description",  
"created" : "2020-06-02T10:45:09Z",  
"modified" : 1,  
"is_delete" : false  
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;  
import com.huaweicloud.sdk.projectman.v4.*;  
import com.huaweicloud.sdk.projectman.v4.model.*;  
  
public class CreateCustomfieldsSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ProjectManClient client = ProjectManClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))  
            .build();  
        CreateCustomfieldsRequest request = new CreateCustomfieldsRequest();  
        CreateCustomfieldV1Req body = new CreateCustomfieldV1Req();  
        body.withScrumType("Story");  
        body.withMemo("描述");  
        body.withOptions("0792b7d26cb74bd799cbae80063dc74a");  
        body.withType("text");  
        body.withName("test");  
        request.withBody(body);  
        try {  
            CreateCustomfieldsResponse response = client.createCustomfields(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
        }  
    }  
}
```

```
        System.out.println(e.getStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateCustomfieldsRequest()
        request.body = CreateCustomfieldV1Req(
            scrum_type="Story",
            memo="描述",
            options="0792b7d26cb74bd799cbae80063dc74a",
            type="text",
            name="test"
        )
        response = client.create_customfields(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
```

```
sk := os.Getenv("CLOUD_SDK_SK")

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := projectman.NewProjectManClient(
    projectman.ProjectManClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateCustomfieldsRequest{
    request.Body = &model.CreateCustomfieldV1Req{
        ScrumType: "Story",
        Memo: "描述",
        Options: "0792b7d26cb74bd799cbae80063dc74a",
        Type: "text",
        Name: "test",
    }
}
response, err := client.CreateCustomfields(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.2 上传图片

功能介绍

上传图片

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/img

表 4-7 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<p>参数解释：项目的32位uuid，项目唯一标识，可以通过ListProjectsV4接口查询项目列表</p> <p>约束限制：正则表达式，<code>^[A-Za-z0-9]{32}\$</code></p> <p>取值范围：字符串，最小长度：32，最大长度：32</p> <p>默认取值：不涉及</p>

请求参数

表 4-8 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	<p>参数解释：用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。</p> <p>约束限制：不涉及</p> <p>取值范围：最小长度：10，最大长度：32768</p> <p>默认取值：不涉及</p>

表 4-9 FormData 参数

参数	是否必选	参数类型	描述
file	是	File	<p>参数解释：图片文件</p> <p>约束限制：图片大小上限 5M</p> <p>取值范围：不涉及</p> <p>默认取值：不涉及</p>

响应参数

状态码： 200

表 4-10 响应 Body 参数

参数	参数类型	描述
img_id	String	图片id
img_url	String	图片url v1改成v3作为下载图片请求

状态码： 400

表 4-11 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-12 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Post https://{endpoint}/v2/99be3c6242d74ad180292a689f95d666/img
```

```
Post https://{endpoint}/v2/99be3c6242d74ad180292a689f95d666/img
```

响应示例

状态码： 200

OK

```
{  
  "img_id" : 1,  
  "img_url" : "/v1/upload/0792b7d26cb74bd799cbac80063dc74a/202006/1591339394752881984.gif"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;
```



```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class UploadIssuelmgSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        UploadIssuelmgRequest request = new UploadIssuelmgRequest();
        try {
            UploadIssuelmgResponse response = client.uploadIssuelmg(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
```

```
.with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
.build()

try:
    request = UploadIssueImgRequest()
    response = client.upload_issue_img(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UploadIssueImgRequest{}
    response, err := client.UploadIssueImg(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK

状态码	描述
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.3 获取工作项完成率

功能介绍

获取工作项的完成率

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issue-completion-rate

表 4-13 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-14 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码： 200

表 4-15 响应 Body 参数

参数	参数类型	描述
issue_completion_rates	Array of issue_completion_rates objects	不同类型的工作项完成率
total	Integer	总数

表 4-16 issue_completion_rates

参数	参数类型	描述
issue_status	issue_status object	工作项不同状态下的数量
tracker_id	Integer	工作项类型id,1需求,2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story

表 4-17 issue_status

参数	参数类型	描述
closed_num	Integer	已关闭的工作项
new_num	Integer	新建的工作项
process_num	Integer	进行中的工作项数目
rejected_num	Integer	已经拒绝的工作项
solved_num	Integer	已经解决的工作项
test_num	Integer	测试中的工作项

状态码： 400

表 4-18 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-19 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

GET https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb1/issue-completion-rate

响应示例

状态码： 200

OK

```
{
  "issue_completion_rates": [ {
    "issue_status": {
      "closed_num": 0,
      "new_num": 0,
      "process_num": 0,
      "rejected_num": 0,
      "solved_num": 0,
      "test_num": 0
    },
    "tracker_id": 3
  } ],
  "total": 2
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ShowIssueCompletionRateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
```

```
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
ShowIssueCompletionRateRequest request = new ShowIssueCompletionRateRequest();
try {
    ShowIssueCompletionRateResponse response = client.showIssueCompletionRate(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowIssueCompletionRateRequest()
        response = client.show_issue_completion_rate(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowIssueCompletionRateRequest{}
    response, err := client.ShowIssueCompletionRate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.4 获取指定工作项的评论列表

功能介绍

获取工作项的评论

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issues/{issue_id}/comments

表 4-20 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32
issue_id	是	Integer	工作项id

表 4-21 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	分页索引，偏移量 最小值：0 缺省值：0
limit	否	Integer	每页显示的条数,最大显示100条 最小值：1 最大值：100 缺省值：10

请求参数

表 4-22 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码： 200

表 4-23 响应 Body 参数

参数	参数类型	描述
total	Integer	评论总数
comments	Array of IssueCommentV4 objects	评论列表

表 4-24 IssueCommentV4

参数	参数类型	描述
comment	String	评论内容
id	Integer	评论id
created_time	String	评论时间
timestamp	String	评论时间戳
user	CommentUserV4 object	

表 4-25 CommentUserV4

参数	参数类型	描述
user_num_id	Integer	发表评论用户id
user_name	String	发表评论用户名称
nick_name	String	发表评论用户昵称

状态码： 400

表 4-26 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-27 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Get https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb1/issues/1232495/comments

Get https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb1/issues/1232495/comments

响应示例

状态码： 200

OK

```
{
  "total": 3,
  "comments": [ {
    "comment": "<p>打的费</p>",
    "id": 2777430,
    "created_time": "2019-10-18",
    "timestamp": "1684220287000",
    "user": {
      "user_num_id": 4091,
      "user_name": "demo_user_name",
      "nick_name": "name"
    }
  }
  ]
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.5 按用户查询工时（单项目）

功能介绍

按用户查询工时（单项目）

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/work-hours

表 4-28 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-29 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

表 4-30 请求 Body 参数

参数	是否必选	参数类型	描述
user_ids	否	Array of strings	查询的用户id列表
work_hours_types	否	String	工时类型，以逗号分隔,21:研发设计,22:后端开发,23:前端开发(Web),24:前端开发(小程序),25:前端开发(App),26:测试验证,27:缺陷修复,28:UI设计,29:会议,30:公共事务,31:培训,32:研究,33:其它,34:调休请假
work_hours_dates	否	String	工时日期，以逗号分隔，年-月-日
begin_time	否	String	工时开始日期，年-月-日

参数	是否必选	参数类型	描述
end_time	否	String	工时结束日期, 年-月-日
offset	是	Integer	偏移量,offset是limit的整数倍, limit=10,offset=0,10,20... 最小值: 0 缺省值: 0
limit	是	Integer	每页显示数量, 每页最多显示 100条 最小值: 1 最大值: 100 缺省值: 10

响应参数

状态码: 200

表 4-31 响应 Body 参数

参数	参数类型	描述
work_hours	Array of work_hours objects	工时列表
total	Integer	总数

表 4-32 work_hours

参数	参数类型	描述
project_name	String	项目名称
nick_name	String	用户昵称
user_id	String	用户id
user_name	String	用户名
work_date	String	工时日期
work_hours_num	String	工时花费
summary	String	工时内容
work_hours_type_name	String	工时类型

参数	参数类型	描述
issue_id	Integer	工作项id
issue_type	String	工作项类型
subject	String	工作项标题
created_time	String	工作项创建时间
closed_time	String	工作项结束时间
work_hours_created_time	String	工时创建时间
work_hours_updated_time	String	工时更新时间

状态码： 400

表 4-33 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-34 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Post https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb2/work-hours

```
{
  "limit": 1,
  "offset": 0,
  "end_time": "2020-08-14"
}
```

响应示例

状态码： 200

ok

```
{
  "work_hours": [ {
    "project_name": "demo",
    "nick_name": "demo",
    "user_id": "09d75cdd9a00d44d2f17c01875483f26",
    "user_name": "userName",
    "work_date": "2020-02-19",
    "work_hours_num": "2.0",
    "summary": "summaryXXXXX",
    "work_hours_type_name": "后端开发",
    "issue_id": 123456,
    "issue_type": "Story",
    "subject": "title",
    "created_time": "2019-12-17",
    "closed_time": "2019-12-20",
    "work_hours_created_time": "2019-12-17 09:00:00",
    "work_hours_updated_time": "2019-12-20 15:00:00"
  } ],
  "total": 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ShowProjectWorkHoursSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowProjectWorkHoursRequest request = new ShowProjectWorkHoursRequest();
        ShowProjectWorkHoursRequestBody body = new ShowProjectWorkHoursRequestBody();
        body.withLimit(1);
        body.withOffset(0);
        body.withEndTime("2020-08-14");
        request.withBody(body);
        try {
            ShowProjectWorkHoursResponse response = client.showProjectWorkHours(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
```

```
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowProjectWorkHoursRequest()
        request.body = ShowProjectWorkHoursRequestBody(
            limit=1,
            offset=0,
            end_time="2020-08-14"
        )
        response = client.show_project_work_hours(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
```

```
// In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := projectman.NewProjectManClient(
    projectman.ProjectManClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowProjectWorkHoursRequest{}
endTimeShowProjectWorkHoursRequestBody := "2020-08-14"
request.Body = &model.ShowProjectWorkHoursRequestBody{
    Limit: int32(1),
    Offset: int32(0),
    EndTime: &endTimeShowProjectWorkHoursRequestBody,
}
response, err := client.ShowProjectWorkHours(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	ok
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.6 按用户查询工时（多项目）

功能介绍

按用户查询工时（多项目）

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/work-hours

请求参数

表 4-35 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度： 10 最大长度： 32768

表 4-36 请求 Body 参数

参数	是否必选	参数类型	描述
project_ids	否	Array of strings	查询的项目id列表
user_ids	否	Array of strings	查询的用户id列表
work_hours_types	否	String	工时类型，以逗号分隔,21:研发设计,22:后端开发,23:前端开发(Web),24:前端开发(小程序),25:前端开发(App),26:测试验证,27:缺陷修复,28:UI设计,29:会议,30:公共事务,31:培训,32:研究,33:其它,34:调休请假
work_hours_dates	否	String	工时日期，以逗号分隔，年-月-日
begin_time	否	String	工时开始日期，年-月-日
end_time	否	String	工时结束日期，年-月-日
offset	是	Integer	偏移量,offset是limit的整数倍, limit=10,offset=0,10,20... 最小值： 0
limit	是	Integer	每页显示数量，每页最多显示100条 最小值： 1 最大值： 100

响应参数

状态码： 200

表 4-37 响应 Body 参数

参数	参数类型	描述
work_hours	Array of work_hours objects	工时列表
total	Integer	总数

表 4-38 work_hours

参数	参数类型	描述
project_name	String	项目名称
nick_name	String	用户昵称
user_id	String	用户id
user_name	String	用户名
work_date	String	工时日期
work_hours_number	String	工时花费
summary	String	工时内容
work_hours_type_name	String	工时类型
issue_id	Integer	工作项id
issue_type	String	工作项类型
subject	String	工作项标题
created_time	String	工作项创建时间
closed_time	String	工作项结束时间
work_hours_created_time	String	工时创建时间
work_hours_updated_time	String	工时更新时间

状态码： 400

表 4-39 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-40 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Post https://{endpoint}/v4/projects/work-hours

```
{
  "limit" : 1,
  "offset" : 0,
  "end_time" : "2020-08-14",
  "project_ids" : [ "e2da96a5d2c845e284f0ad47f8ca8cb2", "a2da96a5d2c845e284f0ad47f8ca8cef" ]
}
```

响应示例

状态码： 200

ok

```
{
  "work_hours" : [ {
    "project_name" : "项目名称",
    "nick_name" : "用户昵称",
    "user_id" : "09d75cdd9a00d44d2f17c01875483f26",
    "user_name" : "用户名",
    "work_date" : "2020-02-19",
    "work_hours_num" : "2.0",
    "summary" : "做了XXXXX",
    "work_hours_type_name" : "后端开发",
    "issue_id" : 123456,
    "issue_type" : "Story",
    "subject" : "工作项标题",
    "created_time" : "2019-12-17",
    "closed_time" : "2019-12-20",
    "work_hours_created_time" : "2019-12-17 09:00:00",
    "work_hours_updated_time" : "2019-12-20 15:00:00"
  } ],
  "total" : 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

import java.util.List;
import java.util.ArrayList;

public class ListProjectWorkHoursSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ListProjectWorkHoursRequest request = new ListProjectWorkHoursRequest();
        ListProjectWorkHoursRequestBody body = new ListProjectWorkHoursRequestBody();
        List<String> listbodyProjectIds = new ArrayList<>();
        listbodyProjectIds.add("e2da96a5d2c845e284f0ad47f8ca8cb2");
        listbodyProjectIds.add("a2da96a5d2c845e284f0ad47f8ca8cef");
        body.withLimit(1);
        body.withOffset(0);
        body.withEndTime("2020-08-14");
        body.withProjectIds(listbodyProjectIds);
        request.withBody(body);
        try {
            ListProjectWorkHoursResponse response = client.listProjectWorkHours(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
```

```
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListProjectWorkHoursRequest()
        listProjectIdsbody = [
            "e2da96a5d2c845e284f0ad47f8ca8cb2",
            "a2da96a5d2c845e284f0ad47f8ca8cef"
        ]
        request.body = ListProjectWorkHoursRequestBody(
            limit=1,
            offset=0,
            end_time="2020-08-14",
            project_ids=listProjectIdsbody
        )
        response = client.list_project_work_hours(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
```

```
request := &model.ListProjectWorkHoursRequest{}
var listProjectIdsbody = []string{
    "e2da96a5d2c845e284f0ad47f8ca8cb2",
    "a2da96a5d2c845e284f0ad47f8ca8cef",
}
endTimeListProjectWorkHoursRequestBody := "2020-08-14"
request.Body = &model.ListProjectWorkHoursRequestBody{
    Limit: int32(1),
    Offset: int32(0),
    EndTime: &endTimeListProjectWorkHoursRequestBody,
    ProjectIds: &listProjectIdsbody,
}
response, err := client.ListProjectWorkHours(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	ok
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.7 添加指定工作项工时

功能介绍

添加指定工作项工时

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/issues/{issue_id}/work-hours

表 4-41 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<p>参数解释: 项目的32位uuid, 项目唯一标识, 可以通过 ListProjectsV4接口查询项目列表</p> <p>约束限制: 正则表达式, <code>^[A-Za-z0-9]{32}\$</code></p> <p>取值范围: 字符串, 最小长度: 32, 最大长度: 32</p> <p>默认取值: 不涉及</p>
issue_id	是	Integer	<p>参数解释: 工作项id, 可通过 ListIssuesV4接口高级查询工作项</p> <p>约束限制: 正则表达式, <code>[0-9]{1,10}\$</code></p> <p>取值范围: 最小长度: 1, 最大长度: 10</p> <p>默认取值: 不涉及</p>

请求参数

表 4-42 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	<p>参数解释: 用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。</p> <p>约束限制: 不涉及</p> <p>取值范围: 最小长度: 10, 最大长度: 32768</p> <p>默认取值: 不涉及</p>

表 4-43 请求 Body 参数

参数	是否必选	参数类型	描述
start_date	是	String	<p>参数解释: 工时开始日期, 年-月-日</p> <p>约束限制: 正则表达式, $\wedge \backslash d\{4\}-\backslash d\{2\}-\backslash d\{2\}\\$</p> <p>取值范围: 匹配格式为 YYYY-MM-DD 的日期字符串, MM 是两位数的月份限制取值 01-12, DD是两位数的日期限制取值01-31</p> <p>默认取值: 不涉及</p>
due_date	是	String	<p>参数解释: 工时结束日期, 年-月-日</p> <p>约束限制: 正则表达式, $\wedge \backslash d\{4\}-\backslash d\{2\}-\backslash d\{2\}\\$</p> <p>取值范围: 匹配格式为 YYYY-MM-DD 的日期字符串, MM 是两位数的月份限制取值 01-12, DD是两位数的日期限制取值01-31</p> <p>默认取值: 不涉及</p>
work_hours	是	Double	<p>参数解释: 工时总数 (若工时日期范围包含多天, 单日工时将设为“工时总数/天数”)</p> <p>约束限制: 不涉及</p> <p>取值范围: 最小值: 0.0, 最大值: 100000000</p> <p>默认取值: 不涉及</p>
work_hours_type_id	否	Integer	<p>参数解释: 工时类型id (项目预设工时类型id及名称对照: 21:研发设计, 22:后端开发, 23:前端开发(Web), 24:前端开发(小程序), 25:前端开发(App), 26:测试验证, 27:缺陷修复, 28:UI设计, 29:会议, 30:公共事务, 31:培训, 32:研究, 33:其它, 34:调休请假)</p> <p>约束限制: 不涉及</p> <p>取值范围: 最小值: 21, 最大值: 34</p> <p>默认取值: 不涉及</p>

响应参数

状态码： 200

表 4-44 响应 Body 参数

参数	参数类型	描述
added_work_hours	Array of added_work_hours objects	已添加的工时列表

表 4-45 added_work_hours

参数	参数类型	描述
work_hours_id	String	工时id
issue_id	Integer	添加工时对应的工作项id
user_name	String	工时添加用户的用户名
user_nick_name	String	工时添加用户的用户昵称
work_date	String	工时日期 年-月-日
work_hours	Double	工时数
work_hours_type_name	String	工时类型名称 (项目预设工时类型id及名称对照: 21:研发设计, 22:后端开发, 23:前端开发(Web), 24:前端开发(小程序), 25:前端开发(App), 26:测试验证, 27:缺陷修复, 28:UI设计, 29:会议, 30:公共事务, 31:培训, 32:研究, 33:其它, 34:调休请假)

状态码： 400

表 4-46 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-47 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Post https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb2/issues/123456/work-hours

```
{
  "start_date": "2023-05-15",
  "due_date": "2023-05-15",
  "work_hours": 2.0,
  "work_hours_type_id": 21
}
```

响应示例

状态码: 200

ok

```
{
  "added_work_hours": [ {
    "work_hours_id": "7da14cf9f3c111eda5a5fa163ec9f6c1",
    "issue_id": 123456,
    "user_name": "user_name",
    "user_nick_name": "O_O",
    "work_date": "2023-05-15",
    "work_hours": "2.0",
    "work_hours_type_name": "研发设计"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class AddIssueWorkHoursSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    }
}
```

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
AddIssueWorkHoursRequest request = new AddIssueWorkHoursRequest();
AddIssueWorkHoursRequestBody body = new AddIssueWorkHoursRequestBody();
body.withWorkHoursTypeId(21);
body.withWorkHours((double)2.0);
body.withDueDate("2023-05-15");
body.withStartDate("2023-05-15");
request.withBody(body);
try {
    AddIssueWorkHoursResponse response = client.addIssueWorkHours(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = AddIssueWorkHoursRequest()
        request.body = AddIssueWorkHoursRequestBody(
            work_hours_type_id=21,
            work_hours=2.0,
            due_date="2023-05-15",
            start_date="2023-05-15"
        )
        response = client.add_issue_work_hours(request)
```

```
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.AddIssueWorkHoursRequest{}
    workHoursTypeIdAddIssueWorkHoursRequestBody := int32(21)
    request.Body = &model.AddIssueWorkHoursRequestBody{
        WorkHoursTypeId: &workHoursTypeIdAddIssueWorkHoursRequestBody,
        WorkHours: float64(2.0),
        DueDate: "2023-05-15",
        StartDate: "2023-05-15",
    }
    response, err := client.AddIssueWorkHours(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	ok

状态码	描述
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.8 查询项目下的工时类型

功能介绍

查询项目下的工时类型

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/work-hours-type

表 4-48 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目id

表 4-49 Query 参数

参数	是否必选	参数类型	描述
limit	否	Integer	每页显示的数量，默认显示10条，最多显示100条 最小值：1 最大值：100 缺省值：10
offset	否	Integer	分页索引，偏移量offset是limit的整数倍， limit=10,offset=0,10,20... 最小值：0 缺省值：0

参数	是否必选	参数类型	描述
status	否	Integer	工时类型状态，支持按状态筛选查询，置空时查询所有工时类型，1表示查询启用的工时类型，2表示查询未启用的工时类型

请求参数

表 4-50 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-51 响应 Body 参数

参数	参数类型	描述
work_hours_types	Array of WorkHoursType objects	工时类型列表
total	Integer	总数

表 4-52 WorkHoursType

参数	参数类型	描述
id	Integer	工时类型id
name	String	工时类型名称
status	Integer	工时类型状态，1表示启用中，2表示未启用

状态码：400

表 4-53 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-54 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Get https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb2/work-hours-type

Get https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb2/work-hours-type?limit=1&offset=0

响应示例

状态码： 200

ok

```
{
  "work_hours_types": [ {
    "id": 21,
    "name": "研发设计",
    "status": 1
  }, {
    "id": 22,
    "name": "后端开发",
    "status": 1
  }, {
    "id": 23,
    "name": "前端开发(Web)",
    "status": 1
  }, {
    "id": 24,
    "name": "前端开发(小程序)",
    "status": 1
  }, {
    "id": 25,
    "name": "前端开发(App)",
    "status": 1
  }, {
    "id": 26,
    "name": "测试验证",
    "status": 1
  }, {
    "id": 27,
    "name": "缺陷修复",
    "status": 1
  }
]
```

```
  }],  
  "total": 7  
}
```

状态码

状态码	描述
200	ok
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.9 获取工作项历史记录

功能介绍

获取工作项历史记录

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issue/{issue_id}/records

表 4-55 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32
issue_id	是	Integer	工作项id

表 4-56 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	分页索引, 偏移量,offset是limit的整数倍, limit=10,offset=0,10,20... 最小值: 0 缺省值: 0
limit	否	Integer	每页显示的数量,每页最多显示100条 最小值: 1 最大值: 1000 缺省值: 10

请求参数

表 4-57 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取(响应消息头中X-Subject-Token的值)。 最小长度: 10 最大长度: 32768

响应参数

状态码: 200

表 4-58 响应 Body 参数

参数	参数类型	描述
id	Integer	操作记录id (已废弃)
created_time	Long	创建时间 (已废弃)
records	Array of IssueRecordV4 objects	历史记录
total	Integer	操作记录总数

表 4-59 IssueRecordV4

参数	参数类型	描述
id	Integer	操作记录id
created_time	Long	操作记录创建时间
user	user object	工作项操作的用户
details	Array of details objects	操作的记录

表 4-60 user

参数	参数类型	描述
user_num_id	Integer	用户数字id
user_name	String	登录名
nick_name	String	昵称
user_id	String	用户32位的uuid

表 4-61 details

参数	参数类型	描述
property	String	操作属性
old_value	String	上次的记录
new_value	String	当前值
operation	String	操作
id	Integer	操作记录的id
name	String	操作的字段

状态码： 400

表 4-62 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401**表 4-63** 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Get https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb1/issue/3458/records
```

```
Get https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb1/issue/3458/records
```

响应示例**状态码： 200**

OK

```
{
  "id": 1221,
  "created_time": "1597116605000",
  "records": [ {
    "id": 1234,
    "created_time": 1617764919000,
    "user": {
      "user_num_id": 4091,
      "user_name": "demo_user_name",
      "nick_name": "aaa",
      "user_id": "a360371833bf4c558f796fd707b44daf"
    },
    "details": [ {
      "property": 1,
      "old_value": 1,
      "new_value": 2,
      "operation": "批量编辑",
      "id": 3597,
      "name": "标题"
    } ]
  } ],
  "total": 1
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.10 创建工作项

功能介绍

创建工作项

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/issue

表 4-64 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	参数解释： 项目的32位uuid，项目唯一标识，可以通过ListProjectsV4接口查询项目列表 约束限制： 正则表达式， <code>^[A-Za-z0-9]{32}\$</code> 取值范围： 字符串，最小长度： 32 ，最大长度： 32 默认取值： 不涉及

请求参数

表 4-65 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	参数解释： 用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 约束限制： 不涉及 取值范围： 最小长度： 10 ，最大长度： 32768 默认取值： 不涉及

表 4-66 请求 Body 参数

参数	是否必选	参数类型	描述
actual_work_hours	否	Double	参数解释 : 实际工时, 实际工时由详细工时自动累加和子工作项自动卷积计算 约束限制 : 不涉及 取值范围 : 最小长度: 0 默认取值 : 不涉及
assigned_id	否	Integer	参数解释 : 处理人数字id, 可通过ListProjectMembersV4接口获取指定项目的成员用户列表, 确认特定项目成员的用户数字id 约束限制 : 正则表达式, $\backslash d + \$$ 取值范围 : 最小值 0 默认取值 : 不涉及
begin_time	否	String	参数解释 : 开始时间, 年-月-日 约束限制 : 正则表达式, $^ [1 - 9] \backslash d \{ 3 \} - (0 ? [1 - 9] 1 [0 - 2]) - (0 [1 - 9] [1 - 2] [0 - 9] 3 [0 - 1]) \$$ 取值范围 : 匹配格式为 YYYY-MM-DD 的日期字符串, MM是两位数的月份限制取值01-12, DD是两位数的日期限制取值01-31 默认取值 : 不涉及
description	否	String	参数解释 : 工作项描述 约束限制 : 可以为空 取值范围 : 最小长度 0 默认取值 : 不涉及
developer_id	否	Integer	参数解释 : 开发人员数字id, 可通过ListProjectMembersV4接口获取指定项目的成员用户列表, 确认特定项目成员的用户数字id 约束限制 : 正则表达式, $\backslash d + \$$ 取值范围 : 最小值 0 默认取值 : 不涉及

参数	是否必选	参数类型	描述
domain_id	否	Integer	<p>参数解释: 领域id, 14 '性能', 15 '功能', 16 '可靠性' 17 '网络安全' 18 '可维护性' 19 '其他DFX' 20 '可用性'</p> <p>约束限制: 正则表达式, $\backslash d + \\$</p> <p>取值范围: 最小值 14 最大值 20</p> <p>默认取值: 不涉及</p>
done_ratio	否	Integer	<p>参数解释: 工作项完成度</p> <p>约束限制: 0 '0%' (输入数字0代表完成度为0%的工作项), 10 '10%', 20 '20%', 30 '30%', 40 '40%', 50 '50%', 60 '60%', 70 '70%' 80 '80%', 90 '90%', 100 '100%'</p> <p>取值范围: 最小值 0 最大值 100</p> <p>默认取值: 不涉及</p>
end_time	否	String	<p>参数解释: 结束时间, 年-月-日</p> <p>约束限制: 正则表达式, $\wedge [1-9] \backslash d \{3\} - (0?[1-9] 1[0-2]) - (0[1-9] 1[1-2][0-9] 3[0-1]) \\$</p> <p>取值范围: 匹配格式为 YYYY-MM-DD 的日期字符串, MM是两位数的月份限制取值 01-12, DD是两位数的日期限制取值 01-31</p> <p>默认取值: 不涉及</p>
expected_work_hours	否	Double	<p>参数解释: 预计工时</p> <p>约束限制: 可以为空</p> <p>取值范围: 最小长度 0</p> <p>默认取值: 不涉及</p>
iteration_id	否	Integer	<p>参数解释: 迭代id, 可通过 ListProjectIterationsV4接口获取指定项目的迭代列表</p> <p>约束限制: 正则表达式, $\backslash d + \\$</p> <p>取值范围: 最小长度 0</p> <p>默认取值: 不涉及</p>

参数	是否必选	参数类型	描述
module_id	否	Integer	参数解释: 模块id, 设置-工作项设置-模块设置内用户可新建模块 约束限制: 正则表达式, $\backslash d + \$$ 取值范围: 最小长度 0 默认取值: 不涉及
name	是	String	参数解释: 标题 约束限制: 可以为空 取值范围: 最小长度 0 默认取值: 不涉及
parent_issue_id	否	Integer	参数解释: 父工作项的id 约束限制: 创建子工作项时必须填, 父工作项的类型tracker_id不能为2 Task, 3 Bug 取值范围: 最小长度 0 默认取值: 不涉及
priority_id	是	Integer	参数解释: 优先级, 约束限制: 正则表达式, $\backslash d + \$$ 取值范围: 1 低, 2 中, 3 高, 默认取值: 不涉及
severity_id	否	Integer	参数解释: 重要程度, 约束限制: 正则表达式, $\backslash d + \$$ 取值范围: 10 关键, 11 重要, 12 一般, 13 提示, 默认取值: 不涉及
status_id	否	Integer	参数解释: 状态id, 约束限制: 正则表达式, $\backslash d + \$$ 取值范围: 新建 1, 进行中 2, 已解决 3, 测试中 4, 已关闭 5, 已拒绝 6 默认取值: 不涉及
tracker_id	是	Integer	参数解释: 工作项类型, 约束限制: Epic只能为 Feature 的父工作项类型; Feature只能为 Story的父工作项类型; Story只能为任务/Task,3缺陷/Bug的父; 取值范围: 2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story; 默认取值: 不涉及

参数	是否必选	参数类型	描述
new_custom_fields	否	Array of NewCustomField objects	参数解释: 用户自定义字段, 可以通过ListIssueCustomFields查询Scrum工作项自定义字段 约束限制: 不涉及 取值范围: 不涉及 默认取值: 不涉及

表 4-67 NewCustomField

参数	是否必选	参数类型	描述
custom_field	否	String	参数解释: 自定义字段, 可以通过ListIssueCustomFields查询Scrum工作项自定义字段中返回的custom_field值获取 约束限制: 可以为空 取值范围: 不涉及 默认取值: 不涉及
field_name	否	String	参数解释: 自定义字段页面显示的含义, 可以通过ListIssueCustomFields查询Scrum工作项自定义字段中返回的name值获取 约束限制: 可以为空 取值范围: 不涉及 默认取值: 不涉及
value	否	String	参数解释: 自定义属性对应的值, 多个值以英文逗号区分开, 可以通过ListIssueCustomFields查询Scrum工作项自定义字段中返回的options值获取 约束限制: 可以为空 取值范围: 不涉及 默认取值: 不涉及

响应参数

状态码: 200

表 4-68 响应 Body 参数

参数	参数类型	描述
actual_work_hours	Double	实际工时
assigned_cc_user	Array of IssueUser objects	抄送人
assigned_user	IssueUser object	处理人
begin_time	String	开始时间, 年-月-日
creator	IssueUser object	创建人
custom_fields	Array of CustomField objects	自定义属性值, 不建议使用, 建议参考 new_custom_fields 字段
new_custom_fields	Array of NewCustomField objects	自定义属性值
developer	IssueUser object	开发人员
domain	domain object	id 领域, 14 '性能', 15 '功能', 16 '可靠性', 17 '网络安全', 18 '可维护性', 19 '其他DFX', 20 '可用性',
done_ratio	Integer	工作项完成度
end_time	String	结束时间, 年-月-日
expected_work_hours	Double	预计工时
id	Integer	工作项id
project	IssueProjectResponseV4 object	工作项所属项目信息
iteration	iteration object	迭代
module	module object	模块
parent_issue	parent_issue object	父工作项信息
priority	priority object	工作项优先级, 1 '低', 2 '中', 3 '高'

参数	参数类型	描述
severity	severity object	工作项重要程度
status	status object	工作项状态
tracker	tracker object	工作项类型 2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story

表 4-69 IssueUser

参数	参数类型	描述
user_id	String	用户32位uuid, 用户唯一标识, 需要通过IAM获取
user_num_id	Integer	用户数字id
id	Integer	用户id
name	String	用户名
nick_name	String	用户昵称

表 4-70 CustomField

参数	参数类型	描述
name	String	自定义字段名
value	String	自定义字段对应的值
new_name	String	自定义字段名

表 4-71 NewCustomField

参数	参数类型	描述
custom_field	String	自定义字段
field_name	String	自定义字段名称
value	String	自定义字段对应的值, 多个值以英文逗号区分开

表 4-72 domain

参数	参数类型	描述
id	Integer	领域id, 14 '性能', 15 '功能', 16 '可靠性', 17 '网络安全', 18 '可维护性', 19 '其他DFX', 20 '可用性',
name	String	领域, '性能', '功能', '可靠性', '网络安全', '可维护性', '其他DFX', '可用性',

表 4-73 IssueProjectResponseV4

参数	参数类型	描述
project_id	String	项目32位uuid, 项目唯一标识
project_name	String	项目名称
project_num_id	Integer	项目数字id

表 4-74 iteration

参数	参数类型	描述
id	Integer	迭代id
name	String	迭代名

表 4-75 module

参数	参数类型	描述
id	Integer	模块id
name	String	模块

表 4-76 parent_issue

参数	参数类型	描述
id	Integer	父工作项id
name	String	父工作项标题

表 4-77 priority

参数	参数类型	描述
id	Integer	优先级id, 1 '低', 2 '中', 3 '高'
name	String	优先级, '低', '中', '高'

表 4-78 severity

参数	参数类型	描述
id	Integer	重要程度id, 10 关键, 11 重要, 12 一般, 13 提示
name	String	重要程度, 关键, 重要, 一般, 提示

表 4-79 status

参数	参数类型	描述
id	Integer	状态id, 新建 1, 进行中 2, 已解决 3, 测试中 4, 已关闭 5, 已拒绝 6,
name	String	状态名称, 新建, 进行中, 已解决, 测试中, 已关闭, 已拒绝

表 4-80 tracker

参数	参数类型	描述
id	Integer	工作项类型id, 2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story
name	String	工作项类型名称, 任务/Task, 缺陷/Bug, Epic, Feature, Story

状态码: 400

表 4-81 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码: 401

表 4-82 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

POST https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issue

```
{
  "parent_issue_id": 1679822,
  "end_time": "2018-07-05",
  "name": "test",
  "begin_time": "2018-1-31",
  "description": "",
  "assigned_id": 4091,
  "developer_id": 4091,
  "tracker_id": 3,
  "status_id": 1,
  "priority_id": 2,
  "severity_id": 10,
  "iteration_id": 11302238
}
```

响应示例

状态码: 200

OK

```
{
  "actual_work_hours": 0,
  "assigned_cc_user": [ {
    "id": 4091,
    "name": "user_name_demo",
    "nick_name": "nick_name_demo"
  } ],
  "assigned_user": {
    "id": 4091,
    "name": "user_name_demo",
    "nick_name": "nick_name_demo"
  },
  "begin_time": "2020-09-12",
  "creator": {
    "id": 4091,
    "name": "user_name_demo",
    "nick_name": "nick_name_demo"
  },
  "custom_fields": [ {
    "name": "customField16",
    "value": "多选框",
    "new_name": "custom_field16"
  } ],
  "new_custom_fields": [ {
    "custom_field": "custom_field16",
    "field_name": "field_name",
    "value": "select1,select2"
  } ],
  "developer": {
    "id": 4091,
    "name": "user_name_demo",
    "nick_name": "nick_name_demo"
  }
}
```

```
},
"domain" : {
  "id" : 14,
  "name" : "性能"
},
"done_ratio" : 0,
"end_time" : "2020-09-19",
"expected_work_hours" : 9,
"id" : 4567,
"project" : {
  "project_id" : "526cefde62004de2b62b5e8dd2c2b3af",
  "project_name" : "testDemo",
  "project_num_id" : 78953
},
"iteration" : {
  "id" : 456,
  "name" : "iteration_1"
},
"module" : {
  "id" : 456,
  "name" : "Project"
},
"parent_issue" : {
  "id" : 123,
  "name" : "paren_issue_demo"
},
"priority" : {
  "id" : 1,
  "name" : "低"
},
"severity" : {
  "id" : 12,
  "name" : "一般"
},
"status" : {
  "id" : 1,
  "name" : "新建"
},
"tracker" : {
  "id" : 7,
  "name" : "story"
}
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class CreateIssueV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
    }
}
```

```
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
CreateIssueV4Request request = new CreateIssueV4Request();
CreateIssueRequestV4 body = new CreateIssueRequestV4();
body.withTrackerId(3);
body.withStatusId(1);
body.withSeverityId(10);
body.withPriorityId(2);
body.withParentIssueId(1679822);
body.withName("test");
body.withIterationId(11302238);
body.withEndTime("2018-07-05");
body.withDeveloperId(4091);
body.withDescription("");
body.withBeginTime("2018-1-31");
body.withAssignedId(4091);
request.withBody(body);
try {
    CreateIssueV4Response response = client.createIssueV4(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()
```

```
try:
    request = CreateIssueV4Request()
    request.body = CreateIssueRequestV4(
        tracker_id=3,
        status_id=1,
        severity_id=10,
        priority_id=2,
        parent_issue_id=1679822,
        name="test",
        iteration_id=11302238,
        end_time="2018-07-05",
        developer_id=4091,
        description="",
        begin_time="2018-1-31",
        assigned_id=4091
    )
    response = client.create_issue_v4(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateIssueV4Request{
        statusIdCreateIssueRequestV4:= int32(1)
        severityIdCreateIssueRequestV4:= int32(10)
        parentIdCreateIssueRequestV4:= int32(1679822)
        iterationIdCreateIssueRequestV4:= int32(11302238)
        endTimeCreateIssueRequestV4:= "2018-07-05"
        developerIdCreateIssueRequestV4:= int32(4091)
        descriptionCreateIssueRequestV4:= ""
        beginTimeCreateIssueRequestV4:= "2018-1-31"
        assignedIdCreateIssueRequestV4:= int32(4091)
        request.Body = &model.CreateIssueRequestV4{
            TrackerId: int32(3),
```



```
StatusId: &statusIdCreateIssueRequestV4,  
SeverityId: &severityIdCreateIssueRequestV4,  
PriorityId: int32(2),  
ParentIssueId: &parentIssueIdCreateIssueRequestV4,  
Name: "test",  
IterationId: &iterationIdCreateIssueRequestV4,  
EndTime: &endTimeCreateIssueRequestV4,  
DeveloperId: &developerIdCreateIssueRequestV4,  
Description: &descriptionCreateIssueRequestV4,  
BeginTime: &beginTimeCreateIssueRequestV4,  
AssignedId: &assignedIdCreateIssueRequestV4,  
}  
response, err := client.CreateIssueV4(request)  
if err == nil {  
    fmt.Printf("%v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.11 查询项目的工作项

功能介绍

工作项类型id, 分页参数, 创建时间查询项目的工作项

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issues

表 4-83 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<p>参数解释: 项目的32位uuid, 项目唯一标识, 可以通过 ListProjectsV4接口查询项目列表</p> <p>约束限制: 正则表达式, <code>^[A-Za-z0-9]{32}\$</code></p> <p>取值范围: 字符串, 最小长度: 32, 最大长度: 32</p> <p>默认取值: 不涉及</p>

表 4-84 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	<p>参数解释: 偏移量 从0开始</p> <p>约束限制: 正则表达式, <code>^[0-9]{0,10}\$</code></p> <p>取值范围: 最小值: 0</p> <p>默认取值: 0</p>
limit	否	Integer	<p>参数解释: 每页数量</p> <p>约束限制: 不涉及</p> <p>取值范围: 最小值: 1 最大值: 100</p> <p>默认取值: 100</p>
tracker_id	否	Integer	<p>参数解释: 工作项类型,</p> <p>约束限制: 任务/Task,3缺陷/Bug,5Epic,6Feature,7Story;</p> <p>取值范围: 最小值 2 最大值 7</p> <p>默认取值: 不涉及</p>
created_time_interval	否	String	<p>参数解释: 创建工作项的时间(查询的起始时间,查询的结束时间)</p> <p>约束限制: 正则表达式, <code>^([0-9]{0,12})?(,)?([0-9]{0,12})?\$</code></p> <p>取值范围: 不涉及</p> <p>默认取值: 不涉及</p>

参数	是否必选	参数类型	描述
updated_time_interval	否	String	<p>参数解释：更新工作项的时间(查询的起始时间,查询的结束时间)</p> <p>约束限制：正则表达式, ^([1-9][0-9]{0,12})?(,)?([1-9][0-9]{0,12})?\$\$</p> <p>取值范围：不涉及</p> <p>默认取值：不涉及</p>

请求参数

表 4-85 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	<p>参数解释：用户Token。通过调用IAM服务获取用户Token接口获取(响应消息头中X-Subject-Token的值)。</p> <p>约束限制：不涉及</p> <p>取值范围：最小长度: 10, 最大长度: 32768</p> <p>默认取值：不涉及</p>

响应参数

状态码: 200

表 4-86 响应 Body 参数

参数	参数类型	描述
issues	Array of IssueItemSfv 4 objects	工作项信息
total	Integer	工作项总数

表 4-87 IssueItemSfV4

参数	参数类型	描述
actual_work_hours	Double	实际工时
assigned_user	IssueUser object	处理人信息
author	IssueUser object	作者信息
begin_time	Long	工作项开始时间
closed_time	Long	关闭工作项的时间
created_time	Long	创建时间 年-月-日 时:分:秒
custom_feilds	Array of CustomFeild Record objects	自定义字段信息
developer	IssueUser object	开发人员信息
discover_version	String	发现问题的版本
end_time	Long	工作项结束时间 年-月-日 时:分:秒
done_ratio	Integer	完成度 示例: 0 '0%' (输入数字0代表完成度为0%的工作项), 10 '10%', 20 '20%', 30 '30%', 40 '40%', 50 '50%', 60 '60%', 70 '70%' 80 '80%', 90 '90%', 100 '100%'
expected_work_hours	Double	预计工时
order	Integer	顺序 最小值: 1 最大值: 100
parent_issue_id	Integer	当前工作项的父工作项的id
release_version	String	发布的版本
root_issue_id	Integer	当前工作项对应根节点工作项id
story_point	story_point object	故事点id (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 分别对应故事点值0, 1/2, 1, 2, 3, 5, 8, 13, 21, 40, 80, 100)

参数	参数类型	描述
domain	domain object	领域信息 14, '性能', 15, '功能', 16, '可靠性' 17, '网络安全' 18, '可维护性' 19, '其他DFX' 20, '可用性'
iteration	iteration object	迭代信息
module	module object	模块信息
priority	priority object	优先级, 可选值: 1 '低', 2 '中', 3 '高'
severity	severity object	重要程度, 13 '提示', 12 '一般', 11 '重要', 10 '关键'
status	status object	工作项状态id, 新建 1, 进行中 2, 已解决 3, 测试中 4, 已关闭 5, 已拒绝 6,
tracker	tracker object	工作项类型, 2任务/Task,3缺陷/ Bug,5Epic,6Feature,7Story
subject	String	工作项标题
updated_time	Long	工作项更新时间 年-月-日 时:分:秒

表 4-88 CustomFeildRecord

参数	参数类型	描述
key	String	自定义字段属性
name	String	自定义字段名称
value	String	自定义字段值

表 4-89 IssueUser

参数	参数类型	描述
user_id	String	用户32位uuid, 用户唯一标识
user_num_id	Integer	用户数字id, 可通过调用ShowCurUserInfo接口获取当前用户信息
id	Integer	用户id
name	String	用户名
nick_name	String	用户昵称

表 4-90 story_point

参数	参数类型	描述
id	Integer	故事点id
name	String	故事点

表 4-91 domain

参数	参数类型	描述
id	Integer	领域id, 14, '性能', 15, '功能', 16, '可靠性' 17, '网络安全' 18, '可维护性' 19, '其他DFX' 20, '可用性'
name	String	领域, '性能', '功能', '可靠性', '网络安全', '可维护性', '其他DFX'

表 4-92 iteration

参数	参数类型	描述
id	Integer	迭代id
name	String	迭代名

表 4-93 module

参数	参数类型	描述
id	Integer	模块id
name	String	模块

表 4-94 priority

参数	参数类型	描述
id	Integer	优先级id, 1 低, 2 中, 3高
name	String	优先级, 低, 中, 高

表 4-95 severity

参数	参数类型	描述
id	Integer	重要程度id, 13 提示, 12 一般, 11 重要, 10 关键
name	String	重要程度, 提示, 一般, 重要, 关键

表 4-96 status

参数	参数类型	描述
id	Integer	状态id, 1 新建, 2 进行中, 3 已解决, 4 测试中, 5 已关闭, 6已拒绝
name	String	状态名称, 新建, 进行中, 已解决, 测试中, 已关闭, 已拒绝

表 4-97 tracker

参数	参数类型	描述
id	Integer	工作项类型id 2 任务/Task, 3 缺陷/Bug, 5 Epic, 6 Feature, 7 Story
name	String	类型名称, 任务/Task, 缺陷/Bug, Epic, Feature, Story

状态码: 400

表 4-98 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码: 401

表 4-99 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Get https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues
```

```
Get https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues
```

响应示例

状态码: 200

OK

```
{
  "issues": [ {
    "actual_work_hours": 10,
    "assigned_user": {
      "id": 4091,
      "name": "demo_user_name",
      "nick_name": "demo"
    },
    "author": {
      "id": 4091,
      "name": "demo_user_name",
      "nick_name": "demo"
    },
    "begin_time": 1590940800000,
    "closed_time": 1590940800000,
    "created_time": 1590940800000,
    "custom_feilds": [ {
      "key": "customField15",
      "name": "single",
      "value": "15Demo"
    } ],
    "developer": {
      "id": 4091,
      "name": "demo_user_name",
      "nick_name": "demo"
    },
    "discover_version": "ProjectMan 8.0.0",
    "end_time": 1590940800000,
    "done_ratio": 0,
    "expected_work_hours": 9,
    "order": 1,
    "parent_issue_id": 123,
    "release_version": "ProjectMan 8.0.0",
    "root_issue_id": 0,
    "story_point": {
      "id": 1,
      "name": "0"
    },
    "domain": {
      "id": 14,
      "name": "性能"
    },
    "iteration": {
      "id": 456,
      "name": "迭代1"
    },
    "module": {
      "id": 456,
      "name": "Project"
    },
    "priority": {
      "id": 1,
      "name": "低"
    }
  } ]
}
```



```
},
"severity": {
  "id": 12,
  "name": "一般"
},
"status": {
  "id": 1,
  "name": "新建"
},
"tracker": {
  "id": 7,
  "name": "story"
},
"subject": "demo",
"updated_time": 1576114296000
}],
"total": 1
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.12 高级查询工作项

功能介绍

根据筛选条件查询工作项

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/issues

表 4-100 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<p>参数解释：项目的32位uuid，项目唯一标识，可以通过ListProjectsV4接口查询项目列表</p> <p>约束限制：正则表达式，<code>^[A-Za-z0-9]{32}\$</code></p> <p>取值范围：字符串，最小长度：32，最大长度：32</p> <p>默认取值：不涉及</p>

请求参数

表 4-101 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	<p>参数解释：用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。</p> <p>约束限制：不涉及</p> <p>取值范围：最小长度：10，最大长度：32768</p> <p>默认取值：不涉及</p>

表 4-102 请求 Body 参数

参数	是否必选	参数类型	描述
assigned_ids	否	Array of integers	<p>参数解释：处理人id，可通过ListProjectMembersV4接口获取指定项目的成员用户列表</p> <p>约束限制：正则表达式，<code>^\d+\$</code></p> <p>取值范围：最小值 0</p> <p>默认取值：不涉及</p>
creator_ids	否	Array of integers	<p>参数解释：创建者id，可通过ListProjectMembersV4接口获取指定项目的成员用户列表</p> <p>约束限制：正则表达式，<code>^\d+\$</code></p> <p>取值范围：最小值 0</p> <p>默认取值：不涉及</p>

参数	是否必选	参数类型	描述
developer_ids	否	Array of integers	参数解释: 开发人id, 可通过 ListProjectMembersV4接口获取指定项目的成员用户列表 约束限制: 正则表达式, $\backslash d + \$$ 取值范围: 最小值 0 默认取值: 不涉及
domain_ids	否	Array of integers	参数解释: 领域id, 14 '性能', 15 '功能', 16 '可靠性' 17 '网络安全' 18 '可维护性' 19 '其他DFX' 20 '可用性' 约束限制: 正则表达式, $\backslash d + \$$ 取值范围: 最小值 14 最大值 20 默认取值: 不涉及
done_ratios	否	Array of integers	参数解释: 工作项完成度 约束限制: 0 '0%' (输入数字0代表完成度为0%的工作项), 10 '10%', 20 '20%', 30 '30%', 40 '40%', 50 '50%', 60 '60%', 70 '70%' 80 '80%', 90 '90%', 100 '100%' 取值范围: 最小值 0 最大值 100 默认取值: 不涉及
iteration_ids	否	Array of integers	参数解释: 迭代id, 可通过 ListProjectIterationsV4接口获取指定项目的迭代列表 约束限制: 正则表达式, $\backslash d + \$$ 取值范围: 最小长度 0 默认取值: 不涉及
limit	否	Integer	参数解释: 每页数量 约束限制: 不涉及 取值范围: 最小值: 1 最大值: 100 默认取值: 100

参数	是否必选	参数类型	描述
offset	否	Integer	<p>参数解释: 偏移量 offset是limit的整数倍数, limit=10,offset=0,10,20...</p> <p>约束限制: 正则表达式, ^ [1-9]\d{0,10}\$</p> <p>取值范围: 最小值: 0</p> <p>默认取值: 0</p>
module_ids	否	Array of integers	<p>参数解释: 模块id, 设置-工作项设置-模块设置内用户可新建模块</p> <p>约束限制: 正则表达式, ^\d+\$</p> <p>取值范围: 最小长度 0</p> <p>默认取值: 不涉及</p>
priority_ids	否	Array of integers	<p>参数解释: 优先级,</p> <p>约束限制: 正则表达式, ^\d+\$</p> <p>取值范围: 1 低, 2 中, 3 高,</p> <p>默认取值: 不涉及</p>
query_type	否	String	<p>参数解释: 查询类型,</p> <p>约束限制: 正则表达式, ^(epic feature backlog s*)\$</p> <p>取值范围: backlog (Story Task Bug), feature, epic</p> <p>默认取值: backlog</p>
severity_ids	否	Array of integers	<p>参数解释: 重要程度,</p> <p>约束限制: 正则表达式, ^\d+\$</p> <p>取值范围: 10 关键, 11 重要, 12 一般, 13 提示,</p> <p>默认取值: 不涉及</p>
status_ids	否	Array of integers	<p>参数解释: 状态id,</p> <p>约束限制: 正则表达式, ^\d+\$</p> <p>取值范围: 新建 1, 进行中 2, 已解决 3, 测试中 4, 已关闭 5, 已拒绝 6</p> <p>默认取值: 不涉及</p>

参数	是否必选	参数类型	描述
story_point_ids	否	Array of integers	<p>参数解释: 故事点id, 故事点是一种基于敏捷的估算工作量方法, 故事点综合了交付Story所付出的所有努力、开发复杂度、风险, 可以简单理解为开发所需要的成本。</p> <p>约束限制: 正则表达式, $\backslash d + \\$</p> <p>取值范围: (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 分别对应故事点值0, 1/2, 1, 2, 3, 5, 8, 13, 21, 40, 80, 100)</p> <p>默认取值: 不涉及</p>
tracker_ids	否	Array of integers	<p>参数解释: 工作项类型,</p> <p>约束限制: 任务/Task,3缺陷/Bug,5Epic,6Feature,7Story;</p> <p>取值范围: 最小值 2 最大值 7</p> <p>默认取值: 不涉及</p>
include_deleted	否	Boolean	<p>参数解释: 查询条件是否包含已删除工作项</p> <p>约束限制: 正则表达式, $\wedge(\text{true} \text{false})\\$</p> <p>取值范围: true 查询的工作项包含已经逻辑删除的, false 查询的工作项不包含已经删除的</p> <p>默认取值: 不涉及</p>
created_time_interval	否	String	<p>参数解释: 创建工作项的时间(查询的起始时间,查询的结束时间)</p> <p>约束限制: 正则表达式, $\wedge([1-9][0-9]\{0,12\})?(,)?([1-9][0-9]\{0,12\})?\\$</p> <p>取值范围: 不涉及</p> <p>默认取值: 不涉及</p>
updated_time_interval	否	String	<p>参数解释: 根据工作项的更新时间查询工作项, (查询的起始时间,查询的结束时间)</p> <p>约束限制: 正则表达式, $\wedge([1-9][0-9]\{0,12\})?(,)?([1-9][0-9]\{0,12\})?\\$</p> <p>取值范围: 不涉及</p> <p>默认取值: 不涉及</p>

参数	是否必选	参数类型	描述
closed_time_interval	否	String	<p>参数解释: 根据工作项的结束时间查询工作项, (查询的起始时间,查询的结束时间)</p> <p>约束限制: 正则表达式, ^([1-9][0-9]{0,12})?(,)?([1-9][0-9]{0,12})?\$</p> <p>取值范围: 不涉及</p> <p>默认取值: 不涉及</p>
custom_fields	否	Array of custom_fields objects	<p>参数解释: 用户自定义字段, 可以通过ListIssueCustomFields查询Scrum工作项自定义字段</p> <p>约束限制: 不涉及</p> <p>取值范围: 不涉及</p> <p>默认取值: 不涉及</p>

表 4-103 custom_fields

参数	是否必选	参数类型	描述
custom_field	否	String	<p>参数解释: 自定义字段, 可以通过ListIssueCustomFields查询Scrum工作项自定义字段中返回的custom_field值获取</p> <p>约束限制: 可以为空</p> <p>取值范围: 不涉及</p> <p>默认取值: 不涉及</p>
value	否	String	<p>参数解释: 自定义属性对应的值, 多个值以英文逗号区分开, 可以通过ListIssueCustomFields查询Scrum工作项自定义字段中返回的options值获取</p> <p>约束限制: 可以为空</p> <p>取值范围: 不涉及</p> <p>默认取值: 不涉及</p>

响应参数

状态码: 200

表 4-104 响应 Body 参数

参数	参数类型	描述
issues	Array of ListIssueItemResponse objects	工作项列表
total	Integer	工作项总数

表 4-105 ListIssueItemResponse

参数	参数类型	描述
actual_work_hours	Double	实际工时, 实际工时由详细工时自动累加和子工作项自动卷积计算
assigned_cc_user	Array of IssueUser objects	抄送人信息
assigned_user	IssueUser object	处理人信息
begin_time	String	预计开始时间, 年-月-日
created_time	String	创建时间 年-月-日 时:分:秒
creator	IssueUser object	创建人信息
custom_fields	Array of CustomField objects	自定义字段信息, 不建议使用, 建议参考 new_custom_fields 字段
new_custom_fields	Array of NewCustomField objects	自定义字段信息
developer	IssueUser object	开发人员信息
domain	domain object	id, 领域 14, '性能', 15, '功能', 16, '可靠性' 17, '网络安全' 18, '可维护性' 19, '其他DFX' 20, '可用性'
done_ratio	Integer	工作项进度值
end_time	String	预计结束时间, 年-月-日
expected_work_hours	Double	预计工时
id	Integer	工作项编号

参数	参数类型	描述
project	IssueProjectResponseV4 object	项目信息
iteration	iteration object	迭代
module	module object	模块
name	String	工作项标题
parent_issue	parent_issue object	当前工作项的父工作项信息
priority	priority object	优先级, 可选值: 1 '低', 2 '中', 3 '高'
order	order object	工作项优先级顺序
severity	severity object	重要程度, 13 '提示', 12 '一般', 11'重要', 10 '关键'
status	status object	工作项状态
release_dev	String	工作项发布版本号
find_release_dev	String	缺陷发现版本号 (仅Bug类型工作项具备该字段)
env	env object	缺陷发现环境 (仅Bug类型工作项具备该字段)
tracker	tracker object	工作项类型 2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story
updated_time	String	更新时间 年-月-日 时:分:秒
closed_time	String	关闭时间 年-月-日 时:分:秒
deleted	Boolean	是否已经删除,true 已经删除, false 未删除

表 4-106 IssueUser

参数	参数类型	描述
user_id	String	用户32位uuid, 用户唯一标识, 需要通过IAM获取
user_num_id	Integer	用户数字id
id	Integer	用户id
name	String	用户名
nick_name	String	用户昵称

表 4-107 CustomField

参数	参数类型	描述
name	String	自定义字段名
value	String	自定义字段名对应的值
new_name	String	自定义字段名

表 4-108 NewCustomField

参数	参数类型	描述
custom_field	String	自定义字段
field_name	String	自定义字段名称
value	String	自定义字段对应的值，多个值以英文逗号区分开

表 4-109 domain

参数	参数类型	描述
id	Integer	领域id
name	String	领域名

表 4-110 IssueProjectResponseV4

参数	参数类型	描述
project_id	String	项目id为uuid，唯一标识，可通过IAM获取
project_name	String	项目名称
project_num_id	Integer	项目数字id

表 4-111 iteration

参数	参数类型	描述
id	Integer	迭代id
name	String	迭代名

表 4-112 module

参数	参数类型	描述
id	Integer	模块id
name	String	模块

表 4-113 parent_issue

参数	参数类型	描述
id	Integer	父工作项id
name	String	父工作项标题

表 4-114 priority

参数	参数类型	描述
id	Integer	优先级id, 1 '低', 2 '中', 3 '高'
name	String	优先级, '低', '中', '高'

表 4-115 order

参数	参数类型	描述
id	Integer	优先级顺序id, 例: 1
name	String	优先级顺序名称, 例 "1"

表 4-116 severity

参数	参数类型	描述
id	Integer	工作项重要程度id, 13 '提示', 12 '一般', 11'重要', 10 '关键'
name	String	工作项重要程度名称, '提示', '一般', '重要', '关键'

表 4-117 status

参数	参数类型	描述
id	Integer	工作项状态id, 新建 1, 进行中 2, 已解决 3, 测试中 4, 已关闭 5, 已拒绝 6
name	String	工作项状态名称, 新建, 进行中, 已解决, 测试中, 已关闭, 已拒绝

表 4-118 env

参数	参数类型	描述
id	Integer	缺陷发现环境id (该字段仅Bug类型工作项拥有)
name	String	缺陷发现环境名称 (该字段仅Bug类型工作项拥有)

表 4-119 tracker

参数	参数类型	描述
id	Integer	工作项类型id, 2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story
name	String	工作项类型名称, 任务/Task, 缺陷/Bug, Epic, Feature, Story

状态码: 400

表 4-120 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码: 401

表 4-121 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
POST https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues
{
  "status_ids": [ 1 ]
}
```

响应示例

状态码: 200

OK

```
{
  "issues": [ {
    "actual_work_hours": 0,
    "assigned_cc_user": [ {
      "id": 4091,
      "name": "demo_user_name",
      "nick_name": "张三"
    } ],
    "assigned_user": {
      "id": 4091,
      "name": "demo_user_name",
      "nick_name": "张三"
    },
    "begin_time": "2020-09-12",
    "created_time": "2020-09-13 14:40:11",
    "creator": {
      "id": 4091,
      "name": "demo_user_name",
      "nick_name": "张三"
    },
    "custom_fields": [ {
      "name": "customField16",
      "value": "多选框",
      "new_name": "custom_field16"
    } ],
    "new_custom_fields": [ {
      "custom_field": "custom_field16",
      "field_name": "field_name",
      "value": "多选框1,多选框2"
    } ],
    "developer": {
      "id": 4091,
      "name": "demo_user_name",
      "nick_name": "张三"
    },
    "domain": {
      "id": 14,
      "name": "性能"
    },
    "done_ratio": 0,
    "end_time": "2020-09-19",
    "expected_work_hours": 9,
    "id": 4567,
    "project": {
      "project_id": "526cefde62004de2b62b5e8dd2c2b3af",
      "project_name": "testDemo",
      "project_num_id": 78953
    },
    "iteration": {
      "id": 456,
      "name": "迭代1"
    }
  } ]
}
```

```
    },
    "module" : {
      "id" : 456,
      "name" : "Project"
    },
    "name" : "国际化整改",
    "parent_issue" : {
      "id" : 123,
      "name" : "提供查询项目成员的功能"
    },
    "priority" : {
      "id" : 1,
      "name" : "低"
    },
    "order" : {
      "id" : 1,
      "name" : "1"
    },
    "severity" : {
      "id" : 12,
      "name" : "一般"
    },
    "status" : {
      "id" : 1,
      "name" : "新建"
    },
    "release_dev" : "0727",
    "find_release_dev" : "0728",
    "env" : {
      "id" : 36,
      "name" : "测试环境"
    },
    "tracker" : {
      "id" : 7,
      "name" : "story"
    },
    "updated_time" : "2020-09-14 14:40:11",
    "closed_time" : "2020-09-15 14:40:11",
    "deleted" : false
  }],
  "total" : 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

import java.util.List;
import java.util.ArrayList;

public class ListIssuesV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
```

```
environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
ListIssuesV4Request request = new ListIssuesV4Request();
ListIssueRequestV4 body = new ListIssueRequestV4();
List<Integer> listbodyStatusIds = new ArrayList<>();
listbodyStatusIds.add(1);
body.withStatusIds(listbodyStatusIds);
request.withBody(body);
try {
    ListIssuesV4Response response = client.listIssuesV4(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListIssuesV4Request()
        listStatusIdsbody = [
            1
        ]
        request.body = ListIssueRequestV4(
            status_ids=listStatusIdsbody
```

```
)  
response = client.list_issues_v4(request)  
print(response)  
except exceptions.ClientRequestException as e:  
    print(e.status_code)  
    print(e.request_id)  
    print(e.error_code)  
    print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
  
    client := projectman.NewProjectManClient(  
        projectman.ProjectManClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.ListIssuesV4Request{}  
    var listStatusIdsbody = []int32{  
        int32(1),  
    }  
    request.Body = &model.ListIssueRequestV4{  
        StatusIds: &listStatusIdsbody,  
    }  
    response, err := client.ListIssuesV4(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.13 批量删除工作项

功能介绍

批量删除工作项

调用方法

请参见[如何调用API](#)。

URI

DELETE /v4/projects/{project_id}/issues

表 4-122 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-123 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

表 4-124 请求 Body 参数

参数	是否必选	参数类型	描述
issue_ids	是	Array of integers	工作项的id 数组长度：1 - 100

响应参数

状态码：400

表 4-125 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码：401

表 4-126 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
DELETE https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues
{
  "issue_ids": [ 1660084, 1660085 ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;
```

```
import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

import java.util.List;
import java.util.ArrayList;

public class BatchDeletelssuesV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        BatchDeletelssuesV4Request request = new BatchDeletelssuesV4Request();
        BatchDeletelssuesRequestV4 body = new BatchDeletelssuesRequestV4();
        List<Integer> listbodyIssueIds = new ArrayList<>();
        listbodyIssueIds.add(1660084);
        listbodyIssueIds.add(1660085);
        body.withIssueIds(listbodyIssueIds);
        request.withBody(body);
        try {
            BatchDeletelssuesV4Response response = client.batchDeletelssuesV4(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
```

example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
ak = __import__('os').getenv("CLOUD_SDK_AK")
sk = __import__('os').getenv("CLOUD_SDK_SK")

credentials = BasicCredentials(ak, sk) \

client = ProjectManClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = BatchDeletelssuesV4Request()
    listssueidsbody = [
        1660084,
        1660085
    ]
    request.body = BatchDeleltelssuesRequestV4(
        issue_ids=listssueidsbody
    )
    response = client.batch_delete_issues_v4(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.BatchDeletelssuesV4Request{}
    var listssueidsbody = []int32{
        int32(1660084),
        int32(1660085),
    }
    request.Body = &model.BatchDeleltelssuesRequestV4{
        Issueids: listssueidsbody,
    }
    response, err := client.BatchDeletelssuesV4(request)
```

```
if err == nil {  
    fmt.Printf("%+v\n", response)  
} else {  
    fmt.Println(err)  
}  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.14 查询工作项详情

功能介绍

查询工作项详情

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issues/{issue_id}

表 4-127 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<p>参数解释: 项目的32位uuid, 项目唯一标识, 可以通过 ListProjectsV4接口查询项目列表</p> <p>约束限制: 正则表达式, <code>^[A-Za-z0-9]{32}\$</code></p> <p>取值范围: 字符串, 最小长度: 32, 最大长度: 32</p> <p>默认取值: 不涉及</p>
issue_id	是	Integer	<p>参数解释: 工作项id, 可通过 ListIssuesV4接口高级查询工作项</p> <p>约束限制: 正则表达式, <code>[0-9]{1,10}\$</code></p> <p>取值范围: 最小长度: 1, 最大长度: 10</p> <p>默认取值: 不涉及</p>

请求参数

表 4-128 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	<p>参数解释: 用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。</p> <p>约束限制: 不涉及</p> <p>取值范围: 最小长度: 10, 最大长度: 32768</p> <p>默认取值: 不涉及</p>

响应参数

状态码: 200

表 4-129 响应 Body 参数

参数	参数类型	描述
actual_work_hours	Double	实际工时, 实际工时由详细工时自动累加和子工作项自动卷积计算
assigned_cc_user	Array of IssueUser objects	抄送人信息
assigned_user	IssueUser object	处理人信息
begin_time	String	预计开始时间, 年-月-日
created_time	String	创建时间 年-月-日 时:分:秒
creator	IssueUser object	创建人信息
custom_fields	Array of CustomField objects	自定义字段, 不建议使用, 建议参考 new_custom_fields 字段
new_custom_fields	Array of IssueDetailCustomField objects	自定义字段信息
developer	IssueUser object	开发人员信息
domain	domain object	id, 领域 14, '性能', 15, '功能', 16, '可靠性' 17, '网络安全' 18, '可维护性' 19, '其他DFX' 20, '可用性'
done_ratio	Integer	工作项完成度
end_time	String	预计结束时间, 年-月-日
expected_work_hours	Double	预计工时
id	Integer	工作项id
project	IssueProjectResponseV4 object	项目信息
iteration	iteration object	迭代
story_point	story_point object	故事点
module	module object	模块
name	String	标题

参数	参数类型	描述
parent_issue	parent_issue object	父工作项
priority	priority object	工作项优先级
severity	severity object	工作项重要程度， 13 '提示', 12 '一般', 11 '重要', 10 '关键'
status	status object	工作项状态
release_dev	String	工作项发布版本号
find_release_dev	String	缺陷发现版本号（仅Bug类型工作项具备该字段）
env	env object	缺陷发现环境（仅Bug类型工作项具备该字段）
tracker	tracker object	工作项类型 2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story
updated_time	String	更新时间 年-月-日 时:分:秒
closed_time	String	关闭时间 年-月-日 时:分:秒
description	String	工作项描述
order	IssueOrder object	
accessories	Array of IssueAccessory objects	附近列表

表 4-130 IssueUser

参数	参数类型	描述
user_id	String	用户uuid
user_num_id	Integer	用户数字id
id	Integer	用户id
name	String	用户名
nick_name	String	昵称

表 4-131 CustomField

参数	参数类型	描述
name	String	自定义属性名
value	String	自定义属性对应的值
new_name	String	自定义属性名

表 4-132 IssueDetailCustomField

参数	参数类型	描述
custom_field	String	自定义字段
field_name	String	自定义字段名称
value	String	自定义属性对应的值，多个值以英文逗号区分开
field_type	String	自定义字段类型，textArea 多行文本，text 单行文本，select 下拉框，number 数字，time_date 日期，checkbox 多选框，radio 单选框
description	String	自定义字段描述

表 4-133 domain

参数	参数类型	描述
id	Integer	领域id
name	String	领域

表 4-134 IssueProjectResponseV4

参数	参数类型	描述
project_id	String	项目id
project_name	String	项目名称
project_num_id	Integer	项目数字id

表 4-135 iteration

参数	参数类型	描述
id	Integer	迭代id
name	String	迭代名

表 4-136 story_point

参数	参数类型	描述
id	Integer	故事点id
name	String	故事点名称

表 4-137 module

参数	参数类型	描述
id	Integer	模块id
name	String	模块

表 4-138 parent_issue

参数	参数类型	描述
id	Integer	父工作项id
name	String	父工作项

表 4-139 priority

参数	参数类型	描述
id	Integer	优先级id
name	String	优先级

表 4-140 severity

参数	参数类型	描述
id	Integer	重要程度id
name	String	重要程度

表 4-141 status

参数	参数类型	描述
id	Integer	状态id
name	String	状态名称

表 4-142 env

参数	参数类型	描述
id	Integer	缺陷发现环境id
name	String	缺陷发现环境名称

表 4-143 tracker

参数	参数类型	描述
id	Integer	类型id
name	String	类型名称

表 4-144 IssueOrder

参数	参数类型	描述
id	Integer	顺序id
name	String	顺序值

表 4-145 IssueAccessory

参数	参数类型	描述
attachment_id	Integer	附件id
issue_id	Integer	工作鞋ID
creator_num_id	Integer	创建者数字ID
created_date	String	附件创建时间
file_name	String	上传时文件名

参数	参数类型	描述
container_type	String	附件id
disk_file_name	String	附件名称
digest	String	附件id
disk_directory	String	附件路径
creator_id	String	创建这用户uuid

状态码： 400

表 4-146 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-147 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues/4568921
```

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues/4568921
```

响应示例

状态码： 200

OK

```
{  
  "actual_work_hours" : 0,  
  "assigned_cc_user" : [ {  
    "id" : 4091,  
    "name" : "demo_user_name",  
    "nick_name" : "bear1"  
  } ],  
}
```

```
"assigned_user" : {
  "id" : 4091,
  "name" : "demo_user_name",
  "nick_name" : "bear1"
},
"begin_time" : "2020-09-12",
"created_time" : "2020-09-13 14:40:11",
"creator" : {
  "id" : 4091,
  "name" : "demo_user_name",
  "nick_name" : "bear1"
},
"custom_fields" : [ {
  "name" : "customField16",
  "value" : "value",
  "new_name" : "custom_field16"
} ],
"new_custom_fields" : [ {
  "custom_field" : "custom_field16",
  "field_name" : "bear1",
  "value" : "bear11,bear12",
  "field_type" : "textArea",
  "description" : "description"
} ],
"developer" : {
  "id" : 4091,
  "name" : "demo_user_name",
  "nick_name" : "bear1"
},
"domain" : {
  "id" : 14,
  "name" : "性能"
},
"done_ratio" : 0,
"end_time" : "2020-09-19",
"expected_work_hours" : 9,
"id" : 4567,
"project" : {
  "project_id" : "526cefde62004de2b62b5e8dd2c2b3af",
  "project_name" : "testDemo",
  "project_num_id" : 78953
},
"iteration" : {
  "id" : 456,
  "name" : "iteeration1"
},
"story_point" : {
  "id" : 3,
  "name" : "1"
},
"module" : {
  "id" : 456,
  "name" : "Project"
},
"name" : "issueDemo",
"parent_issue" : {
  "id" : 123,
  "name" : "parentDemo"
},
"priority" : {
  "id" : 1,
  "name" : "低"
},
"severity" : {
  "id" : 12,
  "name" : "一般"
},
"status" : {
  "id" : 1,
```

```
{
  "name": "新建"
},
"release_dev": "0727",
"find_release_dev": "0728",
"env": {
  "id": 36,
  "name": "测试环境"
},
"tracker": {
  "id": 7,
  "name": "story"
},
"updated_time": "2020-09-14 14:40:11",
"closed_time": "2020-09-15 14:40:11",
"description": "demo",
"order": {
  "id": 1,
  "name": "1"
},
"accessories": [ {
  "attachment_id": 123444,
  "issue_id": 123444,
  "creator_num_id": 123444,
  "created_date": "2021-12-15 15:57:16",
  "file_name": "res.jpg",
  "container_type": "issue",
  "disk_file_name": "f0ef7216-fd05-40d4-8dea-6e6057739bb7.jpg",
  "digest": 123444,
  "disk_directory": 123444,
  "creator_id": "a360371833bf4c558f796fd707b44daf"
} ]
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.15 更新工作项

功能介绍

更新工作项

调用方法

请参见[如何调用API](#)。

URI

PUT /v4/projects/{project_id}/issues/{issue_id}

表 4-148 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32
issue_id	是	Integer	工作项id

请求参数

表 4-149 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

表 4-150 请求 Body 参数

参数	是否必选	参数类型	描述
actual_work_hours	否	Double	实际工时
assigned_id	否	Integer	处理人id,对应用户信息的数字id
begin_time	否	String	开始时间, 年-月-日
description	否	String	描述信息
developer_id	否	Integer	开发者id,对应用户信息的数字id
domain_id	否	Integer	领域id
done_ratio	否	Integer	工作项进度值
end_time	否	String	结束时间, 年-月-日
expected_work_hours	否	Double	预计工时
iteration_id	否	Integer	迭代id
module_id	否	Integer	模块id
name	否	String	标题

参数	是否必选	参数类型	描述
parent_issue_id	否	Integer	父工作项的id
priority_id	否	Integer	优先级, 1 低, 2 中, 3 高,
severity_id	否	Integer	重要程度, 10 关键, 11 重要, 12 一般, 13 提示,
status_id	否	Integer	状态 id, 新建 1, 进行中 2, 已解决 3, 测试中 4, 已关闭 5, 已解决 6,
tracker_id	否	Integer	工作项类型,2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story
new_custom_fields	否	Array of NewCustomField objects	用户自定义字段

表 4-151 NewCustomField

参数	是否必选	参数类型	描述
custom_field	否	String	自定义字段
field_name	否	String	自定义字段名称
value	否	String	自定义属性对应的值, 多个值以英文逗号区分开

响应参数

状态码: 200

表 4-152 响应 Body 参数

参数	参数类型	描述
actual_work_hours	Double	实际工时
assigned_cc_user	Array of IssueUser objects	抄送人
assigned_user	IssueUser object	
begin_time	String	预计开始时间, 年-月-日

参数	参数类型	描述
created_time	String	创建时间 年-月-日 时:分:秒
creator	IssueUser object	
custom_fields	Array of CustomField objects	自定义属性值,不建议使用, 建议参考 new_custom_fields字段
new_custom_fields	Array of NewCustomField objects	自定义属性值
developer	IssueUser object	
domain	domain object	id, 领域 14, '性能', 15, '功能', 16, '可靠性' 17, '网络安全' 18, '可维护性' 19, '其他DFX' 20, '可用性'
done_ratio	Integer	工作项进度值
end_time	String	预计结束时间, 年-月-日
expected_work_hours	Double	预计工时
id	Integer	工作项id
project	IssueProjectResponseV4 object	项目信息
iteration	iteration object	迭代
module	module object	模块
name	String	标题
parent_issue	parent_issue object	父工作项
priority	priority object	工作项优先级
order	order object	工作项优先级顺序
severity	severity object	工作项重要程度
status	status object	工作项状态
release_dev	String	工作项发布版本号
find_release_dev	String	缺陷发现版本号 (仅Bug类型工作项具备该字段)

参数	参数类型	描述
env	env object	缺陷发现环境（仅Bug类型工作项具备该字段）
tracker	tracker object	工作项类型 2任务/Task,3缺陷/ Bug,5Epic,6Feature,7Story
updated_time	String	更新时间 年-月-日 时:分:秒
closed_time	String	关闭时间 年-月-日 时:分:秒

表 4-153 IssueUser

参数	参数类型	描述
user_id	String	用户uuid
user_num_id	Integer	用户数字id
id	Integer	用户id
name	String	用户名
nick_name	String	昵称

表 4-154 CustomField

参数	参数类型	描述
name	String	自定义属性名
value	String	自定义属性对应的值
new_name	String	自定义属性名

表 4-155 NewCustomField

参数	参数类型	描述
custom_field	String	自定义字段
field_name	String	自定义字段名称
value	String	自定义属性对应的值，多个值以英文逗号区分开

表 4-156 domain

参数	参数类型	描述
id	Integer	领域id
name	String	领域

表 4-157 IssueProjectResponseV4

参数	参数类型	描述
project_id	String	项目id
project_name	String	项目名称
project_num_id	Integer	项目数字id

表 4-158 iteration

参数	参数类型	描述
id	Integer	迭代id
name	String	迭代名

表 4-159 module

参数	参数类型	描述
id	Integer	模块id
name	String	模块

表 4-160 parent_issue

参数	参数类型	描述
id	Integer	父工作项id
name	String	父工作项

表 4-161 priority

参数	参数类型	描述
id	Integer	优先级id
name	String	优先级

表 4-162 order

参数	参数类型	描述
id	Integer	优先级顺序id
name	String	优先级顺序名称

表 4-163 severity

参数	参数类型	描述
id	Integer	重要程度id
name	String	重要程度

表 4-164 status

参数	参数类型	描述
id	Integer	状态id
name	String	状态名称

表 4-165 env

参数	参数类型	描述
id	Integer	缺陷发现环境id
name	String	缺陷发现环境名称

表 4-166 tracker

参数	参数类型	描述
id	Integer	类型id
name	String	类型名称

状态码： 400**表 4-167** 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401**表 4-168** 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
PUT https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues/4568921
{
  "status_id" : 2
}
```

响应示例**状态码： 200**

OK

```
{
  "actual_work_hours" : 0,
  "assigned_cc_user" : [ {
    "id" : 4091,
    "name" : " demo_user",
    "nick_name" : "demo"
  } ],
  "assigned_user" : {
    "id" : 4091,
    "name" : " demo_user",
    "nick_name" : "demo"
  },
  "begin_time" : "2020-09-12",
  "created_time" : "2020-09-13 14:40:11",
  "creator" : {
    "id" : 4091,
    "name" : " demo_user",
    "nick_name" : "demo"
  },
  "custom_fields" : [ {
    "name" : "customField16",
    "value" : "select1",
    "new_name" : "custom_field16"
  } ],
}
```

```
"new_custom_fields" : [ {
  "custom_field" : "custom_field16",
  "field_name" : "field_name",
  "value" : "select1,select2"
}],
"developer" : {
  "id" : 4091,
  "name" : "demo_user",
  "nick_name" : "demo"
},
"domain" : {
  "id" : 14,
  "name" : "性能"
},
"done_ratio" : 0,
"end_time" : "2020-09-19",
"expected_work_hours" : 9,
"id" : 4567,
"project" : {
  "project_id" : "526cefde62004de2b62b5e8dd2c2b3af",
  "project_name" : "testDemo",
  "project_num_id" : 78953
},
"iteration" : {
  "id" : 456,
  "name" : "iteration1"
},
"module" : {
  "id" : 456,
  "name" : "Project"
},
"name" : "issue name",
"parent_issue" : {
  "id" : 123,
  "name" : "parent issue name"
},
"priority" : {
  "id" : 1,
  "name" : "低"
},
"severity" : {
  "id" : 12,
  "name" : "一般"
},
"status" : {
  "id" : 1,
  "name" : "新建"
},
"tracker" : {
  "id" : 7,
  "name" : "story"
},
"updated_time" : "2020-09-14 14:40:11",
"closed_time" : "2020-09-15 14:40:11"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
```

```
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class UpdateIssueV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateIssueV4Request request = new UpdateIssueV4Request();
        IssueRequestV4 body = new IssueRequestV4();
        body.withStatusId(2);
        request.withBody(body);
        try {
            UpdateIssueV4Response response = client.updateIssueV4(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
```

```
.build()

try:
    request = UpdateIssueV4Request()
    request.body = IssueRequestV4(
        status_id=2
    )
    response = client.update_issue_v4(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateIssueV4Request{}
    statusIdIssueRequestV4 := int32(2)
    request.Body = &model.IssueRequestV4{
        StatusId: &statusIdIssueRequestV4,
    }
    response, err := client.UpdateIssueV4(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.16 删除工作项

功能介绍

删除工作项

调用方法

请参见[如何调用API](#)。

URI

DELETE /v4/projects/{project_id}/issues/{issue_id}

表 4-169 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32
issue_id	是	Integer	工作项id

请求参数

表 4-170 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度： 10 最大长度： 32768

响应参数

状态码：400

表 4-171 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码：401

表 4-172 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

DELETE https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues/4568921

DELETE https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues/4568921

响应示例

无

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.17 获取子工作项

功能介绍

获取子工作项

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issues/{issue_id}/child

表 4-173 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32
issue_id	是	Integer	工作项id

请求参数

表 4-174 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-175 响应 Body 参数

参数	参数类型	描述
issues	Array of IssueResponseV4 objects	工作项列表
total	Integer	总数

表 4-176 IssueResponseV4

参数	参数类型	描述
actual_work_hours	Double	实际工时
assigned_cc_user	Array of IssueUser objects	抄送人
assigned_user	IssueUser object	
begin_time	String	预计开始时间, 年-月-日
created_time	String	创建时间 年-月-日 时:分:秒
creator	IssueUser object	
custom_fields	Array of CustomField objects	自定义属性值,不建议使用, 建议参考 new_custom_fields字段

参数	参数类型	描述
new_custom_fields	Array of NewCustomField objects	自定义属性值
developer	IssueUser object	
domain	domain object	id, 领域 14, '性能', 15, '功能', 16, '可靠性' 17, '网络安全' 18, '可维护性' 19, '其他DFX' 20, '可用性'
done_ratio	Integer	工作项进度值
end_time	String	预计结束时间, 年-月-日
expected_work_hours	Double	预计工时
id	Integer	工作项id
project	IssueProjectResponseV4 object	项目信息
iteration	iteration object	迭代
module	module object	模块
name	String	标题
parent_issue	parent_issue object	父工作项
priority	priority object	工作项优先级
order	order object	工作项优先级顺序
severity	severity object	工作项重要程度
status	status object	工作项状态
release_dev	String	工作项发布版本号
find_release_dev	String	缺陷发现版本号 (仅Bug类型工作项具备该字段)
env	env object	缺陷发现环境 (仅Bug类型工作项具备该字段)
tracker	tracker object	工作项类型 2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story
updated_time	String	更新时间 年-月-日 时:分:秒
closed_time	String	关闭时间 年-月-日 时:分:秒

表 4-177 IssueUser

参数	参数类型	描述
user_id	String	用户uuid
user_num_id	Integer	用户数字id
id	Integer	用户id
name	String	用户名
nick_name	String	昵称

表 4-178 CustomField

参数	参数类型	描述
name	String	自定义属性名
value	String	自定义属性对应的值
new_name	String	自定义属性名

表 4-179 NewCustomField

参数	参数类型	描述
custom_field	String	自定义字段
field_name	String	自定义字段名称
value	String	自定义属性对应的值，多个值以英文逗号区分开

表 4-180 domain

参数	参数类型	描述
id	Integer	领域id
name	String	领域

表 4-181 IssueProjectResponseV4

参数	参数类型	描述
project_id	String	项目id

参数	参数类型	描述
project_name	String	项目名称
project_num_id	Integer	项目数字id

表 4-182 iteration

参数	参数类型	描述
id	Integer	迭代id
name	String	迭代名

表 4-183 module

参数	参数类型	描述
id	Integer	模块id
name	String	模块

表 4-184 parent_issue

参数	参数类型	描述
id	Integer	父工作项id
name	String	父工作项

表 4-185 priority

参数	参数类型	描述
id	Integer	优先级id
name	String	优先级

表 4-186 order

参数	参数类型	描述
id	Integer	优先级顺序id
name	String	优先级顺序名称

表 4-187 severity

参数	参数类型	描述
id	Integer	重要程度id
name	String	重要程度

表 4-188 status

参数	参数类型	描述
id	Integer	状态id
name	String	状态名称

表 4-189 env

参数	参数类型	描述
id	Integer	缺陷发现环境id
name	String	缺陷发现环境名称

表 4-190 tracker

参数	参数类型	描述
id	Integer	类型id
name	String	类型名称

状态码： 400

表 4-191 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-192 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues/4568921/child
```

```
GET://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues/4568921/child
```

响应示例

状态码： 200

OK

```
{
  "issues": [ {
    "actual_work_hours": 0,
    "assigned_cc_user": [ {
      "id": 4091,
      "name": "demo_user_name",
      "nick_name": "张三"
    } ],
    "assigned_user": {
      "id": 4091,
      "name": "demo_user_name",
      "nick_name": "张三"
    },
    "begin_time": "2020-09-12",
    "created_time": "2020-09-13 14:40:11",
    "creator": {
      "id": 4091,
      "name": "demo_user_name",
      "nick_name": "张三"
    },
    "custom_fields": [ {
      "name": "customField16",
      "value": "多选框",
      "new_name": "custom_field16"
    } ],
    "new_custom_fields": [ {
      "custom_field": "custom_field16",
      "field_name": "field_name",
      "value": "多选框1,多选框2"
    } ],
    "developer": {
      "id": 4091,
      "name": "demo_user_name",
      "nick_name": "张三"
    },
    "domain": {
      "id": 14,
      "name": "性能"
    },
    "done_ratio": 0,
    "end_time": "2020-09-19",
    "expected_work_hours": 9,
    "id": 4567,
    "project": {
      "project_id": "526cefde62004de2b62b5e8dd2c2b3af",
```



```
{
  "project_name": "testDemo",
  "project_num_id": 78953
},
"iteration": {
  "id": 456,
  "name": "迭代1"
},
"module": {
  "id": 456,
  "name": "Project"
},
"name": "国际化整改",
"parent_issue": {
  "id": 123,
  "name": "提供查询项目成员的功能"
},
"priority": {
  "id": 1,
  "name": "低"
},
"severity": {
  "id": 12,
  "name": "一般"
},
"status": {
  "id": 1,
  "name": "新建"
},
"tracker": {
  "id": 7,
  "name": "story"
},
"updated_time": "2020-09-14 14:40:11",
"closed_time": "2020-09-15 14:40:11"
}],
"total": 1
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.18 查询项目下所有工作项的历史记录

功能介绍

查询项目下所有工作项的历史记录

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issues/records

表 4-193 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

表 4-194 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量 从0开始,offset是limit的整数倍, limit=10,offset=0,10,20... 最小值：0 缺省值：0
limit	否	Integer	每页数量 最小1,最大100 最小值：1 最大值：100 缺省值：100
operated_time_interval	否	String	变更工作项的时间(查询的起始时间,查询的结束时间)

请求参数

表 4-195 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-196 响应 Body 参数

参数	参数类型	描述
records	Array of IssueAttrHistoryRecord objects	历史记录
total	Integer	总数

表 4-197 IssueAttrHistoryRecord

参数	参数类型	描述
field_key	String	操作的字段
field_name	String	操作字段的含义
id	Long	历史记录id
issue_id	Long	工作项id
new_value	String	变更后的值,json字符串
old_value	String	变更前的值,json字符串
operated_time	Long	变更的时间
operation	String	操作类型,新建, 修改, 删除
operator	IssueUser object	
property	String	变更的属性

表 4-198 IssueUser

参数	参数类型	描述
user_id	String	用户uuid
user_num_id	Integer	用户数字id
id	Integer	用户id
name	String	用户名
nick_name	String	昵称

状态码: 400

表 4-199 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-200 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues/records

GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues/records

响应示例

状态码： 200

OK

```
{
  "records": [ {
    "field_key": "status",
    "field_name": "状态",
    "id": 789,
    "issue_id": 123,
    "new_value": "{\"id\":\"2\", \"name\":\"开发中\"}",
    "old_value": "{\"id\":\"1\", \"name\":\"创建\"}",
    "operated_time": 1601175640000,
    "operation": "修改",
    "operator": {
      "id": 4091,
      "name": "demo_user_name",
      "nick_name": "张三"
    },
    "property": "attr"
  } ],
  "total": 1
}
```

状态码

状态码	描述
200	OK

状态码	描述
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.19 查询 Scrum 项目的工作项流转配置

功能介绍

查询Scrum项目的工作项流转配置

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issues/workflow/config

表 4-201 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

表 4-202 Query 参数

参数	是否必选	参数类型	描述
tracker_id	是	Integer	工作项类型id [2,3,4,5,6,7]

请求参数

表 4-203 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-204 响应 Body 参数

参数	参数类型	描述
workflows	Array of ScrumStatusFlowVo objects	流转数据

表 4-205 ScrumStatusFlowVo

参数	参数类型	描述
id	String	流转数据的uuid
name	String	状态名称
status_id	Integer	状态id
direct_to	Array of ScrumStatusFlowDirectToVo objects	流转到的数据

表 4-206 ScrumStatusFlowDirectToVo

参数	参数类型	描述
id	String	流转数据的uuid
status_id	Integer	状态id

参数	参数类型	描述
name	String	状态名
enabled	Boolean	是否开启流转

状态码： 400

表 4-207 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-208 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues/workflow/config?
tracker_id=7
```

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues/workflow/config?
tracker_id=7
```

响应示例

状态码： 200

OK

```
{
  "workflows" : [ {
    "id" : "2e645dbab6874a3eafa77ef06b97ea75",
    "name" : "新建",
    "status_id" : 1,
    "direct_to" : [ {
      "id" : "2e645dbab6874a3eafa77ef06b97ea75",
      "status_id" : 2,
      "name" : "新建",
      "enabled" : true
    } ]
  } ]
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.20 细粒度权限用户创建工作项

功能介绍

拥有IAM细粒度权限（projectmanConfig:systemSettingField:set）且在devcloud项目中有创建工作项的权限的用户可以设置工作项的创建者

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/system/issue

表 4-209 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-210 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

表 4-211 请求 Body 参数

参数	是否必选	参数类型	描述
actual_work_hours	否	Double	实际工时
assigned_id	否	Integer	处理人id,对应用户信息的数字id
begin_time	否	String	开始时间, 年-月-日
description	否	String	描述信息
developer_id	否	Integer	开发者id,对应用户信息的数字id
domain_id	否	Integer	id 领域, 14 '性能', 15 '功能', 16 '可靠性' 17 '网络安全' 18 '可维护性' 19 '其他DFX' 20 '可用性'
done_ratio	否	Integer	工作项进度值
end_time	否	String	结束时间, 年-月-日
expected_work_hours	否	Double	预计工时
iteration_id	否	Integer	迭代id
module_id	否	Integer	模块id
name	是	String	标题
parent_issue_id	否	Integer	父工作项的id,创建子工作项时必填, 父工作项的类型tracker_id 不能为2,3
priority_id	是	Integer	优先级, 1 低, 2 中, 3 高,
severity_id	否	Integer	重要程度, 10 关键, 11 重要, 12 一般, 13 提示,

参数	是否必选	参数类型	描述
status_id	否	Integer	状态 id, 新建 1, 进行中 2, 已解决 3, 测试中 4, 已关闭 5, 已拒绝 6,
tracker_id	是	Integer	工作项类型, 2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story; 5只能为 6 的父工作项类型; 6只能为 7 的父工作项类型; 7只能为 2,3的父;
new_custom_fields	否	Array of NewCustomField objects	用户自定义字段
creator	是	Creator object	
custom_fields	否	Array of ScrumCustomField objects	用户自定义字段

表 4-212 NewCustomField

参数	是否必选	参数类型	描述
custom_field	否	String	自定义字段
field_name	否	String	自定义字段名称
value	否	String	自定义属性对应的值, 多个值以英文逗号区分开

表 4-213 Creator

参数	是否必选	参数类型	描述
user_id	是	String	指定工作项的创建者用户id

表 4-214 ScrumCustomField

参数	是否必选	参数类型	描述
name	否	String	自定义字段
value	否	String	自定义字段值

响应参数

状态码： 200

表 4-215 响应 Body 参数

参数	参数类型	描述
actual_work_hours	Double	实际工时
assigned_cc_user	Array of IssueUser objects	抄送人
assigned_user	IssueUser object	
begin_time	String	开始时间, 年-月-日
creator	IssueUser object	
custom_fields	Array of CustomField objects	自定义属性值, 不建议使用, 建议参考 new_custom_fields 字段
new_custom_fields	Array of NewCustomField objects	自定义属性值
developer	IssueUser object	
domain	domain object	id 领域, 14 '性能', 15 '功能', 16 '可靠性', 17 '网络安全', 18 '可维护性', 19 '其他DFX', 20 '可用性',
done_ratio	Integer	工作项进度值
end_time	String	结束时间, 年-月-日
expected_work_hours	Double	预计工时
id	Integer	工作项id
project	IssueProjectResponseV4 object	项目信息
iteration	iteration object	迭代
module	module object	模块
parent_issue	parent_issue object	父工作项

参数	参数类型	描述
priority	priority object	工作项优先级
severity	severity object	工作项重要程度
status	status object	工作项状态
tracker	tracker object	工作项类型 2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story

表 4-216 IssueUser

参数	参数类型	描述
user_id	String	用户uuid
user_num_id	Integer	用户数字id
id	Integer	用户id
name	String	用户名
nick_name	String	昵称

表 4-217 CustomField

参数	参数类型	描述
name	String	自定义属性名
value	String	自定义属性对应的值
new_name	String	自定义属性名

表 4-218 NewCustomField

参数	参数类型	描述
custom_field	String	自定义字段
field_name	String	自定义字段名称
value	String	自定义属性对应的值，多个值以英文逗号区分开

表 4-219 domain

参数	参数类型	描述
id	Integer	领域id
name	String	领域

表 4-220 IssueProjectResponseV4

参数	参数类型	描述
project_id	String	项目id
project_name	String	项目名称
project_num_id	Integer	项目数字id

表 4-221 iteration

参数	参数类型	描述
id	Integer	迭代id
name	String	迭代名

表 4-222 module

参数	参数类型	描述
id	Integer	模块id
name	String	模块

表 4-223 parent_issue

参数	参数类型	描述
id	Integer	父工作项id
name	String	父工作项

表 4-224 priority

参数	参数类型	描述
id	Integer	优先级id
name	String	优先级

表 4-225 severity

参数	参数类型	描述
id	Integer	重要程度id
name	String	重要程度

表 4-226 status

参数	参数类型	描述
id	Integer	状态id
name	String	状态名称

表 4-227 tracker

参数	参数类型	描述
id	Integer	类型id
name	String	类型名称

状态码： 400

表 4-228 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-229 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

POST https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/system/issue

```
{
  "parent_issue_id": 1679822,
  "end_time": "2018-07-05",
  "name": "test",
  "begin_time": "2018-1-31",
  "description": "",
  "assigned_id": 4091,
  "developer_id": 4091,
  "tracker_id": 3,
  "status_id": 1,
  "priority_id": 2,
  "severity_id": 10,
  "iteration_id": 11302238,
  "creator": {
    "user_id": "a360371833bf4c558f796fd707b44daf"
  }
}
```

响应示例

状态码: 200

OK

```
{
  "actual_work_hours": 0,
  "assigned_cc_user": [ {
    "id": 4091,
    "name": "demo_user_name",
    "nick_name": "bear"
  } ],
  "assigned_user": {
    "id": 4091,
    "name": "demo_user_name",
    "nick_name": "bear"
  },
  "begin_time": "2020-09-12",
  "creator": {
    "id": 4091,
    "name": "demo_user_name",
    "nick_name": "bear"
  },
  "custom_fields": [ {
    "name": "customField16",
    "value": "多选框",
    "new_name": "custom_field16"
  } ],
  "new_custom_fields": [ {
    "custom_field": "custom_field16",
    "field_name": "field_name",
    "value": "多选框1,多选框2"
  } ],
  "developer": {
```

```
{
  "id" : 4091,
  "name" : "demo_user_name",
  "nick_name" : "bear1"
},
"domain" : {
  "id" : 14,
  "name" : "性能"
},
"done_ratio" : 0,
"end_time" : "2020-09-19",
"expected_work_hours" : 9,
"id" : 4567,
"project" : {
  "project_id" : "526cefde62004de2b62b5e8dd2c2b3af",
  "project_name" : "testDemo",
  "project_num_id" : 78953
},
"iteration" : {
  "id" : 456,
  "name" : "迭代1"
},
"module" : {
  "id" : 456,
  "name" : "Project"
},
"parent_issue" : {
  "id" : 123,
  "name" : "parentIssue"
},
"priority" : {
  "id" : 1,
  "name" : "低"
},
"severity" : {
  "id" : 12,
  "name" : "一般"
},
"status" : {
  "id" : 1,
  "name" : "新建"
},
"tracker" : {
  "id" : 7,
  "name" : "story"
}
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class CreateSystemIssueV4Solution {

    public static void main(String[] args) {
```



```
// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
CreateSystemIssueV4Request request = new CreateSystemIssueV4Request();
CreateSystemIssueRequestV4 body = new CreateSystemIssueRequestV4();
Creator creatorbody = new Creator();
creatorbody.withUserId("a360371833bf4c558f796fd707b44daf");
body.withAssignedId(4091);
body.withDeveloperId(4091);
body.withSeverityId(10);
body.withParentIssueId(1679822);
body.withName("test");
body.withTrackerId(3);
body.withStatusId(1);
body.withPriorityId(2);
body.withIterationId(11302238);
body.withDescription("");
body.withBeginTime("2018-1-31");
body.withEndTime("2018-07-05");
body.withCreator(creatorbody);
request.withBody(body);
try {
    CreateSystemIssueV4Response response = client.createSystemIssueV4(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")
```

```
credentials = BasicCredentials(ak, sk) \

client = ProjectManClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = CreateSystemIssueV4Request()
    creatorbody = Creator(
        user_id="a360371833bf4c558f796fd707b44daf"
    )
    request.body = CreateSystemIssueRequestV4(
        assigned_id=4091,
        developer_id=4091,
        severity_id=10,
        parent_issue_id=1679822,
        name="test",
        tracker_id=3,
        status_id=1,
        priority_id=2,
        iteration_id=11302238,
        description="",
        begin_time="2018-1-31",
        end_time="2018-07-05",
        creator=creatorbody
    )
    response = client.create_system_issue_v4(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateSystemIssueV4Request{}
    creatorbody := &model.Creator{
```

```
    UserId: "a360371833bf4c558f796fd707b44daf",
  }
  assignedIdCreateSystemIssueRequestV4:= int32(4091)
  developerIdCreateSystemIssueRequestV4:= int32(4091)
  severityIdCreateSystemIssueRequestV4:= int32(10)
  parentIssueIdCreateSystemIssueRequestV4:= int32(1679822)
  statusIdCreateSystemIssueRequestV4:= int32(1)
  iterationIdCreateSystemIssueRequestV4:= int32(11302238)
  descriptionCreateSystemIssueRequestV4:= ""
  beginTimeCreateSystemIssueRequestV4:= "2018-1-31"
  endTimeCreateSystemIssueRequestV4:= "2018-07-05"
  request.Body = &model.CreateSystemIssueRequestV4{
    AssignedId: &assignedIdCreateSystemIssueRequestV4,
    DeveloperId: &developerIdCreateSystemIssueRequestV4,
    SeverityId: &severityIdCreateSystemIssueRequestV4,
    ParentIssueId: &parentIssueIdCreateSystemIssueRequestV4,
    Name: "test",
    TrackerId: int32(3),
    StatusId: &statusIdCreateSystemIssueRequestV4,
    PriorityId: int32(2),
    IterationId: &iterationIdCreateSystemIssueRequestV4,
    Description: &descriptionCreateSystemIssueRequestV4,
    BeginTime: &beginTimeCreateSystemIssueRequestV4,
    EndTime: &endTimeCreateSystemIssueRequestV4,
    Creator: creatorbody,
  }
  response, err := client.CreateSystemIssueV4(request)
  if err == nil {
    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.21 查询当前工作项已经关联的工作项

功能介绍

查询当前工作项已经关联的工作项

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issues/{issue_id}/associated-issues

表 4-230 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32
issue_id	是	Integer	工作项ID

表 4-231 Query 参数

参数	是否必选	参数类型	描述
limit	否	Integer	每页数量
offset	否	Integer	偏移量

请求参数

表 4-232 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-233 响应 Body 参数

参数	参数类型	描述
issues	Array of AssociateIssueDetail objects	关联的工作项列表
total	Integer	总数

表 4-234 AssociateIssueDetail

参数	参数类型	描述
subject	String	工作项标题
issue_id	Integer	工作项ID
project	SimpleProject object	
user	SimpleUser object	
status	StatusVo object	状态

表 4-235 SimpleProject

参数	参数类型	描述
project_name	String	项目名称
project_id	String	项目id

表 4-236 SimpleUser

参数	参数类型	描述
user_num_id	Integer	用户数字id
user_id	String	用户uuid
user_name	String	账号名
nick_name	String	用户昵称

表 4-237 StatusVo

参数	参数类型	描述
id	String	状态id
name	String	状态id对应的值

状态码： 400

表 4-238 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-239 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

GET https://{endpoint}/v4/projects/{project_id}/issues/{issue_id}/associated-issues

响应示例

状态码： 200

```
{
  "issues": [ {
    "subject": "demo",
    "issue_id": 1234,
    "project": {
      "project_name": "demo",
      "project_id": "58d9050550c847c89885c1797a718b0c"
    }
  },
  "user": {
    "user_num_id": 4091,
    "user_id": "a360371833bf4c558f796fd707b44daf",
    "user_name": "demo_user_name_xiongzhitest001",
    "nick_name": "bear"
  },
  "status": {
    "id": "1",
    "name": "未开始"
  }
}
```

```
    }],  
    "total" : 12  
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;  
import com.huaweicloud.sdk.projectman.v4.*;  
import com.huaweicloud.sdk.projectman.v4.model.*;  
  
public class ListAssociatedIssuesSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ProjectManClient client = ProjectManClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ListAssociatedIssuesRequest request = new ListAssociatedIssuesRequest();  
        request.withLimit(<limit>);  
        request.withOffset(<offset>);  
        try {  
            ListAssociatedIssuesResponse response = client.listAssociatedIssues(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8  
  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
```

```
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListAssociatedIssuesRequest()
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_associated_issues(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListAssociatedIssuesRequest{}
    limitRequest := int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest := int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListAssociatedIssues(request)
```



```
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.22 查询当前工作项已经关联的关联 Wiki

功能介绍

查询当前工作项已经关联的关联Wiki

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issues/{issue_id}/associated-wikis

表 4-240 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32
issue_id	是	Integer	工作项ID

表 4-241 Query 参数

参数	是否必选	参数类型	描述
limit	否	Integer	每页数量
offset	否	Integer	偏移量

请求参数

表 4-242 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-243 响应 Body 参数

参数	参数类型	描述
wikis	Array of AttachWikiDetail objects	关联的wiki列表
total	Integer	总数

表 4-244 AttachWikiDetail

参数	参数类型	描述
issue_id	Integer	工作项ID
wiki_title	String	Wiki标题
wiki_author	SimpleUser object	
project	SimpleProject object	

参数	参数类型	描述
created_date	String	创建时间
wiki_id	String	wiki ID
region	String	region值

表 4-245 SimpleUser

参数	参数类型	描述
user_num_id	Integer	用户数字id
user_id	String	用户uuid
user_name	String	账号名
nick_name	String	用户昵称

表 4-246 SimpleProject

参数	参数类型	描述
project_name	String	项目名称
project_id	String	项目id

状态码： 400

表 4-247 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-248 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
GET https://{endpoint}/v4/projects/{project_id}/issues/{issue_id}/associated-wikis
```

响应示例

状态码： 200

```
{
  "wikis": [ {
    "issue_id": 1234,
    "wiki_title": "wikidemo",
    "wiki_author": {
      "user_num_id": 4091,
      "user_id": "a360371833bf4c558f796fd707b44daf",
      "user_name": "demo_user_name_xiongzhitest001",
      "nick_name": "demo"
    },
    "project": {
      "project_name": "demo",
      "project_id": "58d9050550c847c89885c1797a718b0c"
    },
    "created_date": "2021-11-18 19:47:34",
    "wiki_id": "1839097",
    "region": "cn-north-xxx"
  } ],
  "total": "success"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ListAssociatedWikisSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ListAssociatedWikisRequest request = new ListAssociatedWikisRequest();
```

```
request.withLimit(<limit>);
request.withOffset(<offset>);
try {
    ListAssociatedWikisResponse response = client.listAssociatedWikis(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListAssociatedWikisRequest()
        request.limit = <limit>
        request.offset = <offset>
        response = client.list_associated_wikis(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
```

```
// The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := projectman.NewProjectManClient(
    projectman.ProjectManClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListAssociatedWikisRequest{
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    response, err := client.ListAssociatedWikis(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.23 查询当前工作项已经关联的代码提交记录 / 分支创建记录

功能介绍

查询当前工作项已经关联的代码提交记录 / 分支创建记录

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issues/{issue_id}/associated-commits

表 4-249 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<p>参数解释: 项目的32位uuid, 项目唯一标识, 可以通过 ListProjectsV4接口查询项目列表</p> <p>约束限制: 正则表达式, <code>^[A-Za-z0-9]{32}\$</code></p> <p>取值范围: 字符串, 最小长度: 32, 最大长度: 32</p> <p>默认取值: 不涉及</p>
issue_id	是	Integer	<p>参数解释: 工作项id, 可通过 ListIssuesV4接口高级查询工作项</p> <p>约束限制: 正则表达式, <code>[0-9]{1,10}\$</code></p> <p>取值范围: 最小长度: 1, 最大长度: 10</p> <p>默认取值: 不涉及</p>

表 4-250 Query 参数

参数	是否必选	参数类型	描述
type	否	String	<p>参数解释: 查询类型:</p> <p>约束限制: 正则表达式, <code>^(commit branch)\$</code></p> <p>取值范围: commit (提交记录) branch (分支记录)</p> <p>默认取值: 不涉及</p>
limit	否	Integer	<p>参数解释: 每页数量</p> <p>约束限制: 不涉及</p> <p>取值范围: 最小值: 1 最大值: 100</p> <p>默认取值: 100</p>

参数	是否必选	参数类型	描述
offset	否	Integer	参数解释： 偏移量 从0开始 约束限制： 正则表达式, ^ [1-9]\d{0,10}\$ 取值范围： 最小值: 0 默认取值： 0

请求参数

表 4-251 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	参数解释： 用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 约束限制： 不涉及 取值范围： 最小长度: 10 , 最大长度: 32768 默认取值： 不涉及

响应参数

状态码: 200

表 4-252 响应 Body 参数

参数	参数类型	描述
commits	Array of CommitRecordDetail objects	提交记录列表
total	Integer	提交记录总数

表 4-253 CommitRecordDetail

参数	参数类型	描述
repository_id	String	仓库ID
branch_name	String	分支名称

参数	参数类型	描述
commit_id	String	commit id
commit_short_id	String	commit short id
commit_msg	String	commit 信息
commit_url	String	commit URL
user	SimpleUser object	用户信息
type	String	查询的类型
create_date	String	创建时间
update_date	String	更新时间

表 4-254 SimpleUser

参数	参数类型	描述
user_num_id	Integer	用户数字id
user_id	String	用户32位uuid, 用户唯一标识
user_name	String	账号名
nick_name	String	用户昵称

状态码： 400

表 4-255 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-256 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
GET https://{endpoint}/v4/projects/{project_id}/issues/{issue_id}/associated-commits
```

```
GET https://{endpoint}/v4/projects/a360371833bf4c558f796fd707b44daf/issues/122344/associated-commits
```

响应示例

状态码: 200

```
{
  "commits": [ {
    "repository_id": "1865706",
    "branch_name": "master",
    "commit_id": "f608b02311d90f311c41711fdf67003c31305b89",
    "commit_short_id": "f608b023",
    "commit_msg": "demo",
    "commit_url": "https://xxx.xxx/xxx/commit/791877f7d0e0b2032bc5e38425c1245e804e8741?ref=master",
    "user": {
      "user_num_id": 4091,
      "user_id": "a360371833bf4c558f796fd707b44daf",
      "user_name": "demo_user_name_xiongzhitest001",
      "nick_name": "bear"
    },
    "type": "branch",
    "create_date": "2021-11-18 19:47:34",
    "update_date": "2021-11-18 19:47:34"
  } ],
  "total": "success"
}
```

状态码

状态码	描述
200	
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.24 查询关联用例

功能介绍

查询关联用例

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issues/{issue_id}/associate-test-cases

表 4-257 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	参数解释: 项目的32位uuid, 项目唯一标识, 可以通过 ListProjectsV4接口查询项目列表 约束限制: 正则表达式, $^[A-Za-z0-9]{32}$ 取值范围: 字符串, 最小长度: 32, 最大长度: 32 默认取值: 不涉及
issue_id	是	Integer	参数解释: 工作项id, 可通过 ListIssuesV4接口高级查询工作项 约束限制: 正则表达式, $[0-9]{1,10}$ 取值范围: 最小长度: 1, 最大长度: 10 默认取值: 不涉及

表 4-258 Query 参数

参数	是否必选	参数类型	描述
limit	否	Integer	参数解释: 每页数量 约束限制: 不涉及 取值范围: 最小值: 1 最大值: 100 默认取值: 100
offset	否	Integer	参数解释: 偏移量 从0开始 约束限制: 正则表达式, $^[0-9]{1,10}$ 取值范围: 最小值: 0 默认取值: 0

请求参数

表 4-259 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	参数解释： 用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 约束限制： 不涉及 取值范围： 最小长度： 10 ，最大长度： 32768 默认取值： 不涉及

响应参数

状态码： 200

表 4-260 响应 Body 参数

参数	参数类型	描述
test_cases	Array of AssociatedTestCases objects	关联的测试用例列表
total	Integer	关联的测试用例总数

表 4-261 AssociatedTestCase

参数	参数类型	描述
case_id	String	用例ID
case_num	String	用例编号
case_name	String	用例名称
case_level	String	用例等级
status	StatusVo object	状态
creator	SimpleUser object	创新者信息
owner	SimpleUser object	用例归属人信息

参数	参数类型	描述
project	SimpleProject object	项目信息
is_base_line	Integer	是否基线
type	String	用例类型
created_time	Long	创建时间

表 4-262 StatusVo

参数	参数类型	描述
id	String	状态id
name	String	状态id对应的值

表 4-263 SimpleUser

参数	参数类型	描述
user_num_id	Integer	用户数字id
user_id	String	用户32位uuid, 用户唯一标识
user_name	String	用户名
nick_name	String	用户昵称

表 4-264 SimpleProject

参数	参数类型	描述
project_name	String	项目名称
project_id	String	项目32位uuid, 项目唯一标识

状态码: 400

表 4-265 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401**表 4-266 响应 Body 参数**

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

GET https://{endpoint}/v4/projects/{project_id}/issues/{issue_id}/associate-test-cases

GET https://{endpoint}/v4/projects/a360371833bf4c558f796fd707b44daf/issues/1234/associate-test-cases

响应示例

状态码： 200

ok

```
{
  "test_cases": [ {
    "case_id": "36cfaab019dc49cdb53b07ae553f5a",
    "case_num": "1",
    "case_name": "demo",
    "case_level": "2",
    "status": {
      "id": "1",
      "name": "未开始"
    },
    "creator": {
      "user_num_id": 4091,
      "user_id": "a360371833bf4c558f796fd707b44daf",
      "user_name": "demo_user_name_xiongzhitest001",
      "nick_name": "bear1"
    },
    "owner": {
      "user_num_id": 4091,
      "user_id": "a360371833bf4c558f796fd707b44daf",
      "user_name": "demo_user_name_xiongzhitest001",
      "nick_name": "bear1"
    },
    "project": {
      "project_name": "demo",
      "project_id": "58d9050550c847c89885c1797a718b0c"
    },
    "is_base_line": 0,
    "type": "1",
    "created_time": 1233445566000
  } ],
  "total": 0
}
```

状态码

状态码	描述
200	ok

状态码	描述
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.25 查询 Scrum 工作项自定义字段

功能介绍

查询Scrum工作项自定义字段的可选列表,符合custom_fields或者names条件的都返回,2个值都不传, 返回所有的自定义字段列表

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/issues/custom-fields

表 4-267 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id

请求参数

表 4-268 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-269 请求 Body 参数

参数	是否必选	参数类型	描述
custom_fields	否	Array of strings	自定义字段
names	否	Array of strings	自定义字段页面显示的含义
included_not_in_use	否	Boolean	查询结果是否包括未使用的自定义字段，默认仅查询使用中的自定义字段，设为true时查询项目中所有自定义字段

响应参数

状态码： 200

表 4-270 响应 Body 参数

参数	参数类型	描述
datas	Array of IssueCustomField objects	自定义字段返回数据

表 4-271 IssueCustomField

参数	参数类型	描述
name	String	自定义字段
custom_field	String	自定义字段
options	String	自定义字段的可选值，多个值以英文逗号区分

参数	参数类型	描述
type	String	自定义字段类型， textArea 多行文本，只能包含汉字、英文大小写字母、数字、下划线和连接符，不能超过500字符； text 单行文本，只能包含汉字、英文大小写字母、数字、下划线和连接符，不能超过500字符； select 下拉框，只能包含汉字、英文大小写字母、数字、下划线和连接符，每个选项最大长度40个字符，最多可定义60个选项； number 数字，取值范围由用户创建自定义字段时设置； date 日期 精确到年月日， time_date 日期 精确到时分秒，长整型时间戳； checkbox 多选框，只能包含汉字、英文大小写字母、数字、下划线和连接符，每个选项最大长度40个字符，最多可定义60个选项； radio 单选框，只能包含汉字、英文大小写字母、数字、下划线和连接符，每个选项最大长度40个字符，最多可定义60个选项；
tracker_ids	Array of integers	自定义字段支持的工作项类型 2任务/Task,3缺陷/Bug,5Epic,6Feature,7Story
create_time	String	自定义字段创建时间

状态码： 400

表 4-272 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-273 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Post <https://{endpoint}/v4/projects/0792b7d26cb74bd799cbae80063dc74a/issues/custom-fields>

```
{  
  "names": [ "下拉框Demo" ],
```

```
"included_not_in_use" : true  
}
```

响应示例

状态码： 200

OK

```
{  
  "datas" : [{  
    "name" : "selectName",  
    "custom_field" : "custom_field16",  
    "options" : "selectName1,selectName2",  
    "type" : "ratio",  
    "tracker_ids" : [ 7 ],  
    "create_time" : "2023-05-17T09:50:04+08:00"  
  }]  
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;  
import com.huaweicloud.sdk.projectman.v4.*;  
import com.huaweicloud.sdk.projectman.v4.model.*;  
  
import java.util.List;  
import java.util.ArrayList;  
  
public class ListIssueCustomFieldsSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ProjectManClient client = ProjectManClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ListIssueCustomFieldsRequest request = new ListIssueCustomFieldsRequest();  
        ListIssueCustomFieldsRequestBody body = new ListIssueCustomFieldsRequestBody();  
        List<String> listbodyNames = new ArrayList<>();  
        listbodyNames.add("下拉框Demo");  
        body.withIncludedNotInUse(true);  
        body.withNames(listbodyNames);  
        request.withBody(body);  
        try {
```

```
ListIssueCustomFieldsResponse response = client.listIssueCustomFields(request);
System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListIssueCustomFieldsRequest()
        listNamesbody = [
            "下拉框Demo"
        ]
        request.body = ListIssueCustomFieldsRequestBody(
            included_not_in_use=True,
            names=listNamesbody
        )
        response = client.list_issue_custom_fields(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListIssueCustomFieldsRequest{
        var listNamesbody = []string{
            "下拉框Demo",
        }
        includedNotInUseListIssueCustomFieldsRequestBody:= true
        request.Body = &model.ListIssueCustomFieldsRequestBody{
            IncludedNotInUse: &includedNotInUseListIssueCustomFieldsRequestBody,
            Names: &listNamesbody,
        }
    }
    response, err := client.ListIssueCustomFields(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.26 下载图片

功能介绍

下载图片

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/image-file

表 4-274 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id

表 4-275 Query 参数

参数	是否必选	参数类型	描述
image_uri	是	String	图片URI

请求参数

表 4-276 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码： 200

表 4-277 响应 Body 参数

参数	参数类型	描述
-	File	二进制文件

状态码： 400**表 4-278 响应 Body 参数**

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401**表 4-279 响应 Body 参数**

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Get https://{endpoint}/v4/projects/2c42a6d37ccc4197bf0b3dbe91be021a/image-file?image_uri=/v1/upload/2c42a6d37ccc4197bf0b3dbe91be021a/202209/1662003438194761550.png
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class DownloadImageFileSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
    }
}
```

```
ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
DownloadImageFileRequest request = new DownloadImageFileRequest();
request.withImageUri("<image_uri>");
try {
    DownloadImageFileResponse response = client.downloadImageFile(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DownloadImageFileRequest()
        request.image_uri = "<image_uri>"
        response = client.download_image_file(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
```

```
"fmt"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DownloadImageFileRequest{}
    request.ImageUri = "<image_uri>"
    response, err := client.DownloadImageFile(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	二进制文件
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.27 上传工作项附件

功能介绍

上传工作项附件

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/issues/{issue_id}/attachments/upload

表 4-280 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	项目id
issue_id	是	String	工作项id

请求参数

表 4-281 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-282 FormData 参数

参数	是否必选	参数类型	描述
attachment	是	File	附件文件，最大上传大小为50M

响应参数

状态码：200

表 4-283 响应 Body 参数

参数	参数类型	描述
id	Integer	关联id
issue_id	Integer	工作项id
project_id	String	项目id
disk_filename	String	云盘存储名

参数	参数类型	描述
file_name	String	文件名
size	String	文件大小

状态码： 400

表 4-284 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-285 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Post https://{endpoint}/v4/projects/0792b7d26cb74bd799cbae80063dc74a/attachments/upload

响应示例

状态码： 200

```
{
  "id": "1234",
  "issue_id": "1232",
  "project_id": "2c42a6d37ccc4197bf0b3dbe91be021a",
  "disk_filename": "adfadsfasdfa.jpg",
  "file_name": "demo.jpg",
  "size": "1233"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
```

```
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class UploadAttachmentsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        UploadAttachmentsRequest request = new UploadAttachmentsRequest();
        try {
            UploadAttachmentsResponse response = client.uploadAttachments(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
```

```
.build()

try:
    request = UploadAttachmentsRequest()
    response = client.upload_attachments(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UploadAttachmentsRequest{}
    response, err := client.UploadAttachments(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request

状态码	描述
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.28 下载工作项附件

功能介绍

下载工作项附件

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/issues/{issue_id}/attachments/{attachment_id}

表 4-286 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id
issue_id	是	String	工作项id
attachment_id	是	String	附件id

请求参数

表 4-287 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码： 200

表 4-288 响应 Body 参数

参数	参数类型	描述
-	File	二进制文件

状态码： 400

表 4-289 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-290 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Get <https://{endpoint}/v4/projects/f7a67aee4c034fd6944fd5f769e89042/issues/5667740/attachments/72372>

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class DownloadAttachmentSolution {
```

```
public static void main(String[] args) {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
    // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
    // environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running
    // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    String ak = System.getenv("CLOUD_SDK_AK");
    String sk = System.getenv("CLOUD_SDK_SK");

    ICredential auth = new BasicCredentials()
        .withAk(ak)
        .withSk(sk);

    ProjectManClient client = ProjectManClient.newBuilder()
        .withCredential(auth)
        .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
        .build();
    DownloadAttachmentRequest request = new DownloadAttachmentRequest();
    try {
        DownloadAttachmentResponse response = client.downloadAttachment(request);
        System.out.println(response.toString());
    } catch (ConnectionException e) {
        e.printStackTrace();
    } catch (RequestTimeoutException e) {
        e.printStackTrace();
    } catch (ServiceResponseException e) {
        e.printStackTrace();
        System.out.println(e.getHttpStatusCode());
        System.out.println(e.getRequestId());
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DownloadAttachmentRequest()
        response = client.download_attachment(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

```
print(e.error_code)
print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DownloadAttachmentRequest{}
    response, err := client.DownloadAttachment(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	二进制文件
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.29 删除附件

功能介绍

取消工作项与附件关联，如附件为工作项页面上传则删除附件

调用方法

请参见[如何调用API](#)。

URI

DELETE /v4/projects/{project_id}/issues/{issue_id}/attachments/{attachment_id}

表 4-291 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id
issue_id	是	String	工作项id
attachment_id	是	String	附件id

请求参数

表 4-292 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：400

表 4-293 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-294 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
DELETE https://{endpoint}/v4/projects/f7a67aee4c034fd6944fd5f769e89042/issues/5667740/attachments/72372
```

```
DELETE https://{endpoint}/v4/projects/f7a67aee4c034fd6944fd5f769e89042/issues/5667740/attachments/72372
```

响应示例

无

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.30 查询迭代下工作项状态的统计数据（处理人维度）

功能介绍

查询迭代下工作项状态的统计数据（处理人维度）

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/status-statistic

表 4-295 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id

表 4-296 Query 参数

参数	是否必选	参数类型	描述
iteration_id	是	Integer	迭代数字id
tracker_id	是	Integer	自定义字段支持的工作项类型 2 任务/Task,3缺陷/Bug,7Story
status_id	是	Integer	工作项状态数字id

请求参数

表 4-297 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度： 10 最大长度： 32768

响应参数

状态码：200

表 4-298 响应 Body 参数

参数	参数类型	描述
[数组元素]	Array of UserStatusStatistic objects	OK

表 4-299 UserStatusStatistic

参数	参数类型	描述
user	IssueUser object	
item_count	Integer	满足条件的工作项总数
data	Map<String,Integer>	工作项对应状态的统计计数

表 4-300 IssueUser

参数	参数类型	描述
user_id	String	用户uuid
user_num_id	Integer	用户数字id
id	Integer	用户id
name	String	用户名
nick_name	String	昵称

状态码： 400

表 4-301 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-302 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

无

响应示例

状态码： 200

OK

```
[ {
  "user" : {
    "user_id" : "112a642015cf11eca152fa163ec9f6c1",
    "user_num_id" : 5055,
    "name" : "devcloud_devcloud_test_01_test"
  },
  "item_count" : 1,
  "data" : {
    "5874579" : 1
  }
}, {
  "user" : {
    "user_id" : "1139ffb615cf11eca152fa163ec9f6c1",
    "user_num_id" : 5255,
    "name" : "devcloud_devcloud_test_01_test"
  },
  "item_count" : 1,
  "data" : {
    "5874580" : 1
  }
}, {
  "user" : {
    "user_id" : "a360371833bf4c558f796fd707b44daf",
    "user_num_id" : 9078,
    "nick_name" : "sstest5",
    "name" : "devcloud_devcloud_test_01"
  },
  "item_count" : 2,
  "data" : {
    "5874581" : 1,
    "5874584" : 1
  }
}, {
  "user" : {
    "user_id" : "D0XCOQMxQ9Nujj9lybk51lFEQzpdpzMc",
    "user_num_id" : 15437,
    "name" : "devcloud_devcloud_test_01_FederationUser"
  },
  "item_count" : 1,
  "data" : {
    "5874582" : 2
  }
}
]
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;
```

```
public class ListStatusStatisticSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ListStatusStatisticRequest request = new ListStatusStatisticRequest();
        request.withIterationId(<iteration_id>);
        request.withTrackerId(<tracker_id>);
        request.withStatusId(<status_id>);
        try {
            ListStatusStatisticResponse response = client.listStatusStatistic(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListStatusStatisticRequest()
        request.iteration_id = <iteration_id>
```

```
request.tracker_id = <tracker_id>
request.status_id = <status_id>
response = client.list_status_statistic(request)
print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListStatusStatisticRequest{}
    request.IterationId = int32(<iteration_id>)
    request.TrackerId = int32(<tracker_id>)
    request.StatusId = int32(<status_id>)
    response, err := client.ListStatusStatistic(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK

状态码	描述
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.31 获取指定工作项停留时间

功能介绍

获取指定工作项停留时间

调用方法

请参见[如何调用API](#)。

URI

POST /v4/issues/duration

请求参数

表 4-303 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-304 请求 Body 参数

参数	是否必选	参数类型	描述
project_id	否	String	项目uuid
issue_ids	否	Array of strings	工作项id字符串列表

响应参数

状态码： 200

表 4-305 响应 Body 参数

参数	参数类型	描述
fails	Array of strings	计算失败的工作项id,一般指未关闭的工作项
data	Array of data objects	计算失败的工作项id,一般指未关闭的工作项
total_stay_time	Long	停留时间求和 (单位: 秒)
total	Integer	停留时间求和的工作项个数

表 4-306 data

参数	参数类型	描述
id	String	工作项id字符串
stay_time	Long	停留时间 (单位: 秒)

状态码: 400

表 4-307 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码: 401

表 4-308 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
POST https://{endpoint}/v4/issues/duration
```

```
{  
  "project_id": "78deb5314e674251996513b0a790384b",  
}
```

```
"issue_ids" : [ "6330741", "1212123" ]
}
```

响应示例

状态码： 200

OK

```
{
  "fails" : [ "1212123" ],
  "data" : [ {
    "id" : "6330741",
    "stay_time" : 1238172
  } ],
  "total_stay_time" : 1238172,
  "total" : 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

import java.util.List;
import java.util.ArrayList;

public class ListSpecIssueStayTimesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ListSpecIssueStayTimesRequest request = new ListSpecIssueStayTimesRequest();
        ListSpecIssueStayTimesRequestBody body = new ListSpecIssueStayTimesRequestBody();
        List<String> listbodyIssuelds = new ArrayList<>();
        listbodyIssuelds.add("6330741");
        listbodyIssuelds.add("1212123");
        body.withIssuelds(listbodyIssuelds);
        body.withProjectId("78deb5314e674251996513b0a790384b");
        request.withBody(body);
        try {
```

```
ListSpecIssueStayTimesResponse response = client.listSpecIssueStayTimes(request);
System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListSpecIssueStayTimesRequest()
        listssueldsbody = [
            "6330741",
            "1212123"
        ]
        request.body = ListSpecIssueStayTimesRequestBody(
            issue_ids=listssueldsbody,
            project_id="78deb5314e674251996513b0a790384b"
        )
        response = client.list_spec_issue_stay_times(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
```

```
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
  
    client := projectman.NewProjectManClient(  
        projectman.ProjectManClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.ListSpecIssueStayTimesRequest{}  
    var listIssueIdsbody = []string{  
        "6330741",  
        "1212123",  
    }  
    projectIdListSpecIssueStayTimesRequestBody := "78deb5314e674251996513b0a790384b"  
    request.Body = &model.ListSpecIssueStayTimesRequestBody{  
        IssueIds: &listIssueIdsbody,  
        ProjectId: &projectIdListSpecIssueStayTimesRequestBody,  
    }  
    response, err := client.ListSpecIssueStayTimes(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.1.32 高级查询我的待办工作项

功能介绍

高级查询我的待办工作项

调用方法

请参见[如何调用API](#)。

URI

POST /v4/issues

请求参数

表 4-309 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-310 请求 Body 参数

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量,offset是limit的整数倍, limit=10,offset=0,10,20...
limit	否	Integer	每页显示数量 最小值: 1 最大值: 100
subject	否	String	搜索关键词
created_on	否	String	工作项创建时间区间
updated_on	否	String	工作项更新时间区间
closed_on	否	String	工作项结束时间区间
start_date	否	String	工作项预计开始日期区间
due_date	否	String	工作项预计结束日期区间
tracker_id	否	String	工作项类型
status_id	否	String	工作项状态
author_id	否	String	工作项创建人id

参数	是否必选	参数类型	描述
developer_id	否	String	工作项开发人员id
priority_id	否	String	工作项优先级id

响应参数

状态码： 200

表 4-311 响应 Body 参数

参数	参数类型	描述
issue_list	Array of issue_list objects	工作项信息列表 数组长度： 0 - 100
total	Integer	工作项总数

表 4-312 issue_list

参数	参数类型	描述
id	Integer	工作项id
subject	String	工作项标题
parent_issue_id	Integer	父工作项id
parent_issue	parent_issue object	父工作项信息
project	project object	项目信息
release_dev	String	发布版本
find_release_dev	String	发现发布版本
done_ratio	Integer	工作项完成度
expected_work_hours	Double	预计工时
actual_work_hours	Double	实际工时
tracker	tracker object	工作项类型信息
order	order object	优先级顺序信息

参数	参数类型	描述
severity	severity object	重要程度信息
priority	priority object	工作项优先级信息
domain	domain object	领域信息
position	Double	排序数值
module	module object	模块信息
assigned_to	SimpleUserIn object	
author	SimpleUserIn object	
developer	SimpleUserIn object	
closer	SimpleUserIn object	
status	status object	工作项状态信息
deleted	Boolean	是否删除工作项
is_watcher	Boolean	是否关注工作项
closed_flag	Integer	关闭标志
created_on	String	工作项新建时间戳
updated_on	String	工作项更新时间戳
due_date	String	工作项预计结束时间戳

表 4-313 parent_issue

参数	参数类型	描述
id	Integer	父工作项id
subject	String	父工作项标题

表 4-314 project

参数	参数类型	描述
id	Integer	项目id
identifier	String	项目uuid
name	String	项目名称
type	String	项目类型

表 4-315 tracker

参数	参数类型	描述
id	Integer	工作项类型id
name	String	工作项类型名称

表 4-316 order

参数	参数类型	描述
id	Integer	优先级顺序id
name	String	优先级顺序名称

表 4-317 severity

参数	参数类型	描述
id	Integer	重要程度id
name	String	重要程度名称

表 4-318 priority

参数	参数类型	描述
id	Integer	工作项优先级id
name	String	工作项优先级名称

表 4-319 domain

参数	参数类型	描述
id	Integer	领域id
name	String	领域名称

表 4-320 module

参数	参数类型	描述
id	Integer	模块id
name	String	模块名称
path_name	String	模块路径名称

表 4-321 SimpleUserIn

参数	参数类型	描述
id	Integer	用户id
identifier	String	用户uuid
name	String	用户名称

表 4-322 status

参数	参数类型	描述
id	Integer	工作项状态id
name	String	工作项优先级名称

状态码： 400

表 4-323 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-324 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

POST https://{endpoint}/v4/issues

```
{
  "offset" : "0",
  "limit" : "15",
  "status_id" : "5",
  "due_date" : "1682265600000,1682265600000"
}
```

响应示例

状态码: 200

OK

```
{
  "issue_list" : [ {
    "id" : 7220820,
    "subject" : "demo_issue",
    "project" : {
      "name" : "demo_project",
      "id" : 30387805,
      "identifier" : "",
      "type" : "scrum"
    },
    "done_ratio" : 0,
    "expected_work_hours" : 0.0,
    "actual_work_hours" : 0.0,
    "tracker" : {
      "id" : 7,
      "name" : "Story"
    },
    "order" : {
      "id" : 1,
      "name" : "1.0"
    },
    "severity" : {
      "id" : 12,
      "name" : "一般"
    },
    "priority" : {
      "id" : 2,
      "name" : "中"
    },
    "domain" : { },
    "position" : 7220820.0,
    "module" : { },
    "assigned_to" : {
      "id" : 4091,
      "name" : ""
    },
    "author" : {
      "id" : 4091,
      "name" : ""
    }
  },
  ...
]
```

```
"developer" : { },
"closer" : { },
"status" : {
  "id" : 1,
  "name" : "新建"
},
"is_watcher" : false,
"closed_flag" : 0,
"deleted" : false,
"created_on" : "1681783846000",
"updated_on" : "1681783846000",
"due_date" : "1684252800000"
}],
"total" : 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class SearchIssuesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        SearchIssuesRequest request = new SearchIssuesRequest();
        ListWorkTableIssueRequestV4RequestBody body = new ListWorkTableIssueRequestV4RequestBody();
        body.withStatusId("5");
        body.withDueDate("1682265600000,1682265600000");
        body.withLimit(15);
        body.withOffset(0);
        request.withBody(body);
        try {
            SearchIssuesResponse response = client.searchIssues(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
        }
    }
}
```

```
e.printStackTrace();
System.out.println(e.getStatusCode());
System.out.println(e.getRequestId());
System.out.println(e.getErrorCode());
System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = SearchIssuesRequest()
        request.body = ListWorkTableIssueRequestV4RequestBody(
            status_id="5",
            due_date="1682265600000,1682265600000",
            limit=15,
            offset=0
        )
        response = client.search_issues(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
```

```
sk := os.Getenv("CLOUD_SDK_SK")

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := projectman.NewProjectManClient(
    projectman.ProjectManClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.SearchIssuesRequest{}
statusIdListWorkTableIssueRequestV4RequestBody:= "5"
dueDateListWorkTableIssueRequestV4RequestBody:= "1682265600000,1682265600000"
limitListWorkTableIssueRequestV4RequestBody:= int32(15)
offsetListWorkTableIssueRequestV4RequestBody:= int32(0)
request.Body = &model.ListWorkTableIssueRequestV4RequestBody{
    StatusId: &statusIdListWorkTableIssueRequestV4RequestBody,
    DueDate: &dueDateListWorkTableIssueRequestV4RequestBody,
    Limit: &limitListWorkTableIssueRequestV4RequestBody,
    Offset: &offsetListWorkTableIssueRequestV4RequestBody,
}
response, err := client.SearchIssues(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.2 Scrum 项目的迭代

4.2.1 获取指定项目的迭代列表

功能介绍

获取项目迭代

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/iterations

表 4-325 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

表 4-326 Query 参数

参数	是否必选	参数类型	描述
updated_time_interval	否	String	更新迭代的时间(查询的起始时间,查询的结束时间)
include_deleted	否	Boolean	是否包含被删除的迭代,默认false不包含被删除的迭代

请求参数

表 4-327 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-328 响应 Body 参数

参数	参数类型	描述
total	Integer	迭代总数
iterations	Array of iterations objects	迭代信息

表 4-329 iterations

参数	参数类型	描述
description	String	迭代描述
end_time	String	迭代结束时间
id	Integer	迭代id
name	String	迭代标题
begin_time	String	迭代开始时间
status	String	迭代状态
updated_time	Long	迭代更新时间，长整型时间戳
deleted	Boolean	迭代是否已经删除，false, 未删除， true已经删除

状态码： 400

表 4-330 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-331 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Get https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb1/iterations
```

```
Get https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb1/iterations
```

响应示例

状态码: 200

OK

```
{
  "total": 3,
  "iterations": [ {
    "description": "description",
    "end_time": "2019-10-17",
    "id": 9725885,
    "name": "完成html",
    "begin_time": "2019-10-10",
    "status": "0",
    "updated_time": 1640231944000,
    "deleted": false
  } ]
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.2.2 批量删除项目的迭代

功能介绍

批量删除项目的迭代

调用方法

请参见[如何调用API](#)。

URI

DELETE /v4/projects/{project_id}/iterations

表 4-332 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-333 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

表 4-334 请求 Body 参数

参数	是否必选	参数类型	描述
iteration_ids	是	Array of integers	迭代的id 数组长度：1 - 10

响应参数

状态码：400

表 4-335 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码：401

表 4-336 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Delete https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/iterations
{
  "iteration_ids" : [ 11302238, 11470756 ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

import java.util.List;
import java.util.ArrayList;

public class BatchDeleteliterationsV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        BatchDeleteliterationsV4Request request = new BatchDeleteliterationsV4Request();
        BatchDeleteliterationsV4RequestBody body = new BatchDeleteliterationsV4RequestBody();
        List<Integer> listbodyIterationIds = new ArrayList<>();
        listbodyIterationIds.add(11302238);
```

```
listbodyIterationIds.add(11470756);
body.withIterationIds(listbodyIterationIds);
request.withBody(body);
try {
    BatchDeleteIterationsV4Response response = client.batchDeleteIterationsV4(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = BatchDeleteIterationsV4Request()
        listIterationIdsbody = [
            11302238,
            11470756
        ]
        request.body = BatchDeleteIterationsV4RequestBody(
            iteration_ids=listIterationIdsbody
        )
        response = client.batch_delete_iterations_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
```

```
projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.BatchDeleteliterationsV4Request{}
    var listIterationIdsbody = []int32{
        int32(11302238),
        int32(11470756),
    }
    request.Body = &model.BatchDeleteliterationsV4RequestBody{
        IterationIds: listIterationIdsbody,
    }
    response, err := client.BatchDeleteliterationsV4(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.2.3 查看迭代详情

功能介绍

查看迭代详情

调用方法

请参见[如何调用API](#)。

URI

GET /v4/iterations/{iteration_id}

表 4-337 路径参数

参数	是否必选	参数类型	描述
iteration_id	是	Integer	迭代id

请求参数

表 4-338 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度： 10 最大长度： 32768

响应参数

状态码：200

表 4-339 响应 Body 参数

参数	参数类型	描述
begin_time	String	迭代结束时间，年-月-日
charts	Array of Chart objects	燃尽图
closed_total	Integer	已关闭的工单数
created_time	String	迭代创建时间

参数	参数类型	描述
end_time	String	迭代开始时间, 年-月-日
have_task	Boolean	是否有task
iteration_id	Integer	迭代id
name	String	迭代标题
opened_total	Integer	开启的工单数
progress	String	工作进展
total	Integer	工单总数
updated_time	String	迭代更新时间
status	String	迭代的状态,0 未开始, 1 进行中, 2 结束

表 4-340 Chart

参数	参数类型	描述
date	String	统计时间
finished_num	Integer	完成story工单
iteration_id	Integer	迭代id
project_num_id	Integer	项目id
remaining_num	Integer	未完成story数
total	Integer	总story数

状态码: 400

表 4-341 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码: 401

表 4-342 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Get https://{endpoint}/v4/iterations/11225793
```

```
Get https://{endpoint}/v4/iterations/11225793
```

响应示例

状态码： 200

OK

```
{
  "begin_time": "2020-07-12",
  "charts": [
    {
      "date": "2020-09-12",
      "finished_num": 1,
      "iteration_id": 3789,
      "project_num_id": 36587463,
      "remaining_num": 3,
      "total": 4
    }
  ],
  "closed_total": 1,
  "created_time": "2020-06-31 09:10:03",
  "end_time": "2020-07-19",
  "have_task": false,
  "iteration_id": 35789,
  "name": "迭代1",
  "opened_total": 3,
  "progress": "25%",
  "total": 4,
  "updated_time": "2020-06-31 09:10:03",
  "status": "0"
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.2.4 创建 Scrum 项目迭代

功能介绍

创建Scrum项目迭代

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/iteration

表 4-343 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<p>参数解释：项目的32位uuid，项目唯一标识，可以通过ListProjectsV4接口查询项目列表</p> <p>约束限制：正则表达式，<code>^[A-Za-z0-9]{32}\$</code></p> <p>取值范围：字符串，最小长度：32，最大长度：32</p> <p>默认取值：不涉及</p>

请求参数

表 4-344 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	<p>参数解释：用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。</p> <p>约束限制：不涉及</p> <p>取值范围：最小长度：10，最大长度：32768</p> <p>默认取值：不涉及</p>

表 4-345 请求 Body 参数

参数	是否必选	参数类型	描述
begin_time	是	String	参数解释 : 开始时间, 年-月-日 约束限制 : 正则表达式, ^[1-9]\d{3}-(0?[1-9] 1[0-2])-(0[1-9] [1-2][0-9] 3[0-1])\$ 取值范围 : 匹配格式为 YYYY- MM-DD 的日期字符串, MM 是两位数的月份限制取值 01-12, DD是两位数的日期限制 取值01-31 默认取值 : 不涉及
description	否	String	参数解释 : 迭代的描述 约束限制 : 可以为空 取值范围 : 最小长度: 0 最大长 度: 500 默认取值 : 不涉及
end_time	是	String	参数解释 : 结束时间, 年-月-日 约束限制 : 正则表达式, ^[1-9]\d{3}-(0?[1-9] 1[0-2])-(0[1-9] [1-2][0-9] 3[0-1])\$ 取值范围 : 匹配格式为 YYYY- MM-DD 的日期字符串, MM 是两位数的月份限制取值 01-12, DD是两位数的日期限制 取值01-31 默认取值 : 不涉及
name	是	String	参数解释 : 标题 约束限制 : 不涉及 取值范围 : 最小长度: 1 最大长 度: 255 默认取值 : 不涉及

响应参数

状态码: 201

表 4-346 响应 Body 参数

参数	参数类型	描述
id	Integer	迭代id

状态码： 400

表 4-347 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-348 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Post https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/iteration

```
{
  "end_time" : "2018-07-05",
  "name" : "迭代1",
  "begin_time" : "2018-1-31",
  "description" : ""
}
```

响应示例

状态码： 201

OK

```
{
  "id" : 7312121
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class CreateIterationV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateIterationV4Request request = new CreateIterationV4Request();
        CreateIterationRequestV4 body = new CreateIterationRequestV4();
        body.withName("迭代1");
        body.withEndTime("2018-07-05");
        body.withDescription("");
        body.withBeginTime("2018-1-31");
        request.withBody(body);
        try {
            CreateIterationV4Response response = client.createIterationV4(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
```

```
variables and decrypted during use to ensure security.
# In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = __import__('os').getenv("CLOUD_SDK_AK")
sk = __import__('os').getenv("CLOUD_SDK_SK")

credentials = BasicCredentials(ak, sk) \

client = ProjectManClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = CreateIterationV4Request()
    request.body = CreateIterationRequestV4(
        name="迭代1",
        end_time="2018-07-05",
        description="",
        begin_time="2018-1-31"
    )
    response = client.create_iteration_v4(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateIterationV4Request{}
    descriptionCreateIterationRequestV4:= ""
    request.Body = &model.CreateIterationRequestV4{
        Name: "迭代1",
        EndTime: "2018-07-05",
        Description: &descriptionCreateIterationRequestV4,
        BeginTime: "2018-1-31",
    }
}
```

```
response, err := client.CreateIterationV4(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
201	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.2.5 更新 Scrum 项目迭代

功能介绍

更新Scrum项目迭代

调用方法

请参见[如何调用API](#)。

URI

PUT /v4/projects/{project_id}/iterations/{iteration_id}

表 4-349 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32
iteration_id	是	Integer	迭代id

请求参数

表 4-350 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度： 10 最大长度： 32768

表 4-351 请求 Body 参数

参数	是否必选	参数类型	描述
begin_time	是	String	开始时间，年-月-日
description	否	String	描述
end_time	是	String	结束时间，年-月-日
name	是	String	标题
status	否	String	迭代的状态，0 未开始 <--> 1 进行中<--> 2 结束 <--> 1<-->0, 状态不能跨状态更改
over_type	否	String	迭代结束时，工作项的处理（close 所有的工作项关闭，empty 没有关闭的工作项 放在 block里面），status更新为2时需要填写over_type

响应参数

状态码：400

表 4-352 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码：401

表 4-353 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Put https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/iteration/11225793
{
  "end_time" : "2018-07-05",
  "name" : "sttes32s3",
  "begin_time" : "2018-1-31",
  "description" : ""
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class UpdateliterationV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateliterationV4Request request = new UpdateliterationV4Request();
        UpdateliterationRequestV4 body = new UpdateliterationRequestV4();
        body.withName("sttes32s3");
    }
}
```

```
body.withEndTime("2018-07-05");
body.withDescription("");
body.withBeginTime("2018-1-31");
request.withBody(body);
try {
    UpdateIterationV4Response response = client.updateIterationV4(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateIterationV4Request()
        request.body = UpdateIterationRequestV4(
            name="sttes32s3",
            end_time="2018-07-05",
            description="",
            begin_time="2018-1-31"
        )
        response = client.update_iteration_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
```



```
projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateIterationV4Request{
        descriptionUpdateIterationRequestV4:= ""
    }
    request.Body = &model.UpdateIterationRequestV4{
        Name: "sttes32s3",
        EndTime: "2018-07-05",
        Description: &descriptionUpdateIterationRequestV4,
        BeginTime: "2018-1-31",
    }
    response, err := client.UpdateIterationV4(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.2.6 删除项目迭代

功能介绍

删除项目迭代

调用方法

请参见[如何调用API](#)。

URI

DELETE /v4/projects/{project_id}/iterations/{iteration_id}

表 4-354 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32
iteration_id	是	Integer	迭代id

请求参数

表 4-355 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：400

表 4-356 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-357 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Delete https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/iteration/11225793
```

```
Delete https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/iteration/11470756
```

响应示例

无

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.2.7 查看迭代历史记录

功能介绍

查看迭代历史记录

调用方法

请参见[如何调用API](#)。

URI

GET /v4/iterations/{iteration_id}/histories

表 4-358 路径参数

参数	是否必选	参数类型	描述
iteration_id	是	Integer	迭代id

表 4-359 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量 最小值: 0 缺省值: 0
limit	否	Integer	每页数量, 最大为100 最小值: 1 最大值: 100 缺省值: 10

请求参数

表 4-360 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度: 10 最大长度: 32768

响应参数

状态码: 200

表 4-361 响应 Body 参数

参数	参数类型	描述
total	Integer	总数
histories	Array of IterationHistory objects	历史记录

表 4-362 IterationHistory

参数	参数类型	描述
iteration_id	String	迭代ID
project_id	String	项目ID
operator	operator object	变更操作人
operate	String	操作类型
operate_time	String	操作时间
details	Array of details objects	操作详情

表 4-363 operator

参数	参数类型	描述
user_id	String	用户uuid
user_name	String	用户名
nick_name	String	昵称
domain_id	String	租户uuid
domain_name	String	租户名

表 4-364 details

参数	参数类型	描述
operate_field_name	String	变更的字段
new_value	String	操作后的值
old_value	String	操作前的值

状态码： 400

表 4-365 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述

参数	参数类型	描述
error_code	String	错误码

状态码： 401

表 4-366 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Get <https://{endpoint}/v4/iterations/11225793/histories>

响应示例

状态码： 200

OK

```
{
  "total": 1,
  "histories": [ {
    "iteration_id": 123454,
    "project_id": "19274e1adffb4cc0b866ef6dd3ad0100",
    "operator": {
      "user_id": "0a938608ac00f2271f2ec01cccb94c19",
      "user_name": "Banna",
      "nick_name": "红海",
      "domain_id": "9b33eb1d219140fea54d3b5ac689b073",
      "domain_name": "devcloud_test"
    },
    "operate": "NEW",
    "operate_time": "1665473746000",
    "details": [ {
      "operate_field_name": "name",
      "old_value": null,
      "new_value": "test"
    }, {
      "operate_field_name": "description",
      "old_value": null,
      "new_value": null
    }, {
      "operate_field_name": "end_time",
      "old_value": null,
      "new_value": "1664380800000"
    }, {
      "operate_field_name": "status",
      "old_value": null,
      "new_value": "0"
    }, {
      "operate_field_name": "begin_time",
      "old_value": null,
      "new_value": "1663776000000"
    }, {

```

```
"operate_field_name" : "program_version_id",
"old_value" : null,
"new_value" : "-1"
}]
}]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ListIterationHistoriesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ListIterationHistoriesRequest request = new ListIterationHistoriesRequest();
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        try {
            ListIterationHistoriesResponse response = client.listIterationHistories(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8
```

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListIterationHistoriesRequest()
        request.offset = <offset>
        request.limit = <limit>
        response = client.list_iteration_histories(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListIterationHistoriesRequest{}
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
```



```
request.Limit = &limitRequest
response, err := client.ListIterationHistories(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.3 用户信息

4.3.1 获取当前用户信息

功能介绍

获取当前用户信息

调用方法

请参见[如何调用API](#)。

URI

GET /v4/user

请求参数

表 4-367 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-368 响应 Body 参数

参数	参数类型	描述
domain_id	String	租户id
domain_name	String	租户名
user_num_id	Integer	用户数字id
user_id	String	用户id
user_name	String	用户名
nick_name	String	用户昵称
created_time	Long	创建时间
updated_time	Long	更新时间
gender	String	性别
user_type	String	用户类型, User 云用户, Federation 联邦账号,

状态码：400

表 4-369 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-370 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Get https://{endpoint}/v4/user
```

```
Get https://{endpoint}/v4/user
```

响应示例

状态码： 200

OK

```
{
  "domain_id": "4e919d73499648e3b0292cd3cbef806a",
  "domain_name": "demo_user_name",
  "user_num_id": "4060",
  "user_id": "a360371833bf4c558f796fd707b44daf",
  "user_name": "demo_user_name",
  "nick_name": "我的名字要非常非常长才能测出问题",
  "created_time": "1562318865000",
  "updated_time": "1598074854000",
  "gender": "male",
  "user_type": "User"
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.3.2 更新用户昵称

功能介绍

更新用户昵称

调用方法

请参见[如何调用API](#)。

URI

PUT /v4/user

请求参数

表 4-371 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户token

表 4-372 请求 Body 参数

参数	是否必选	参数类型	描述
nick_name	是	String	用户昵称

响应参数

状态码： 400

表 4-373 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-374 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
PUT https://{endpoint}/v4/user
```

```
{  
  "nick_name" : "Tom"  
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;  
import com.huaweicloud.sdk.projectman.v4.*;  
import com.huaweicloud.sdk.projectman.v4.model.*;  
  
public class UpdateNickNameV4Solution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ProjectManClient client = ProjectManClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))  
            .build();  
        UpdateNickNameV4Request request = new UpdateNickNameV4Request();  
        UpdateUserNickNameRequestV4 body = new UpdateUserNickNameRequestV4();  
        body.withNickName("Tom");  
        request.withBody(body);  
        try {  
            UpdateNickNameV4Response response = client.updateNickNameV4(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateNickNameV4Request()
        request.body = UpdateUserNickNameRequestV4(
            nick_name="Tom"
        )
        response = client.update_nick_name_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
```

```
request := &model.UpdateNickNameV4Request{}
request.Body = &model.UpdateUserNickNameRequestV4{
    NickName: "Tom",
}
response, err := client.UpdateNickNameV4(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.3.3 获取当前用户角色

功能介绍

获取用户在项目中的角色

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/user-role

表 4-375 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-376 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-377 响应 Body 参数

参数	参数类型	描述
user_role	Integer	用户角色id 成员角色, -1 项目创建者, 3 项目经理, 4 开发人员, 5 测试经理, 6 测试人员, 7 参与者, 8 浏览者, 9 运维经理

状态码：400

表 4-378 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码：401

表 4-379 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Get https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/user-role
```

响应示例

状态码： 200

OK

```
{
  "user_role" : 3
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ShowCurUserRoleSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ShowCurUserRoleRequest request = new ShowCurUserRoleRequest();
        try {
            ShowCurUserRoleResponse response = client.showCurUserRole(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

```
}  
}
```

Python

```
# coding: utf-8  
  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkprojectman.v4 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = __import__('os').getenv("CLOUD_SDK_AK")  
    sk = __import__('os').getenv("CLOUD_SDK_SK")  
  
    credentials = BasicCredentials(ak, sk) \  
  
    client = ProjectManClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = ShowCurUserRoleRequest()  
        response = client.show_cur_user_role(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
  
    client := projectman.NewProjectManClient(  
        projectman.ProjectManClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())
```

```
request := &model.ShowCurUserRoleRequest{}
response, err := client.ShowCurUserRole(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.3.4 更新子用户昵称

功能介绍

拥有te_admin角色的用户可以更新其他用户的昵称

调用方法

请参见[如何调用API](#)。

URI

PUT /v4/domain/child-users

请求参数

表 4-380 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户token

表 4-381 请求 Body 参数

参数	是否必选	参数类型	描述
users	是	Array of UpdateChildUserNickNameRequestBody objects	修改的用户列表

表 4-382 UpdateChildUserNickNameRequestBody

参数	是否必选	参数类型	描述
nick_name	是	String	用户昵称
user_id	是	String	用户id

响应参数

状态码： 400

表 4-383 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-384 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
PUT https://{endpoint}/v4/domain/child-users
{
  "users": [ {
    "nick_name": "Tom",
    "user_id": "08777d423500d4b01fe5c008d04336ab"
  } ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

import java.util.List;
import java.util.ArrayList;

public class BatchUpdateChildNickNamesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();

        BatchUpdateChildNickNamesRequest request = new BatchUpdateChildNickNamesRequest();
        BatchUpdateChildUserNickNamesRequestBody body = new
BatchUpdateChildUserNickNamesRequestBody();
        List<UpdateChildUserNickNameRequestBody> listbodyUsers = new ArrayList<>();
        listbodyUsers.add(
            new UpdateChildUserNickNameRequestBody()
                .withNickName("Tom")
                .withUserId("08777d423500d4b01fe5c008d04336ab")
        );
        body.withUsers(listbodyUsers);
        request.withBody(body);
        try {
            BatchUpdateChildNickNamesResponse response = client.batchUpdateChildNickNames(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

```
}  
}  
}
```

Python

```
# coding: utf-8  
  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkprojectman.v4 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = __import__('os').getenv("CLOUD_SDK_AK")  
    sk = __import__('os').getenv("CLOUD_SDK_SK")  
  
    credentials = BasicCredentials(ak, sk) \  
  
    client = ProjectManClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = BatchUpdateChildNickNamesRequest()  
        listUsersbody = [  
            UpdateChildUserNickNameRequestBody(  
                nick_name="Tom",  
                user_id="08777d423500d4b01fe5c008d04336ab"  
            )  
        ]  
        request.body = BatchUpdateChildUserNickNamesRequestBody(  
            users=listUsersbody  
        )  
        response = client.batch_update_child_nick_names(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")
```

```
auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := projectman.NewProjectManClient(
    projectman.ProjectManClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.BatchUpdateChildNickNamesRequest{}
var listUsersbody = []model.UpdateChildUserNickNameRequestBody{
    {
        NickName: "Tom",
        UserId: "08777d423500d4b01fe5c008d04336ab",
    },
}
request.Body = &model.BatchUpdateChildUserNickNamesRequestBody{
    Users: listUsersbody,
}
response, err := client.BatchUpdateChildNickNames(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.4 项目成员

4.4.1 主动退出项目

功能介绍

项目成员主动退出项目，项目创建者不能退出

调用方法

请参见[如何调用API](#)。

URI

DELETE /v4/projects/{project_id}/quit

表 4-385 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-386 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：400

表 4-387 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码：401

表 4-388 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Delete https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb1/quit
```

```
Delete https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb1/quit
```

响应示例

无

状态码

状态码	描述
204	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.4.2 获取指定项目的成员用户列表

功能介绍

获取项目成员列表

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/members

表 4-389 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

表 4-390 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	分页索引，偏移量,offset是limit的整数倍, limit=10,offset=0,10,20... 最小值：0 最大值：10000 缺省值：0
limit	否	Integer	每页显示的数量,每页最多显示100条 最小值：1 最大值：100 缺省值：100

请求参数

表 4-391 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-392 响应 Body 参数

参数	参数类型	描述
members	Array of members objects	项目成员列表
total	Integer	总数

表 4-393 members

参数	参数类型	描述
domain_id	String	租户id 最小长度: 32 最大长度: 32
domain_name	String	租户名
user_id	String	用户id
user_name	String	用户名
user_num_id	Integer	创建人numId
role_id	Integer	成员角色, -1 项目创建者, 3 项目经理, 4 开发人员, 5 测试经理, 6 测试人员, 7 参与者, 8 浏览者, 9 运维经理
nick_name	String	用户昵称
role_name	String	用户角色
user_type	String	用户类型, User iam用户, Federation 联邦账号,
forbidden	Integer	是否是禁用账号, 1 禁用账号, 0非禁用账号

状态码: 400

表 4-394 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码: 401

表 4-395 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Get https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb1/members
```

```
Get https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb1/members
```

响应示例

状态码： 200

OK

```
{
  "members": [ {
    "domain_id": "4e919d73499648e3b0292cd3cbef806a",
    "domain_name": "demo_user_name",
    "user_id": "a360371833bf4c558f796fd707b44daf",
    "user_name": "demo_user_name",
    "user_num_id": 4097,
    "role_id": 7,
    "nick_name": "name",
    "role_name": "项目创建者",
    "user_type": "User",
    "forbidden": 0
  } ],
  "total": 1
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.4.3 批量添加项目成员

功能介绍

批量添加项目成员，只能添加和项目创建者同一租户下的成员，不正确的用户id会略过，添加的用户超过权限的，默认角色设置为7

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/members

表 4-396 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-397 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

表 4-398 请求 Body 参数

参数	是否必选	参数类型	描述
users	是	Array of BatchAddMemberRequest V4 objects	添加的用户信息 数组长度：1 - 50

表 4-399 BatchAddMemberRequestV4

参数	是否必选	参数类型	描述
role_id	否	Integer	成员角色，-1 项目创建者，3 项目经理，4 开发人员，5 测试经理，6 测试人员，7 参与者，8 浏览者，9 运维经理

参数	是否必选	参数类型	描述
user_id	是	String	用户32位uuid

响应参数

状态码： 400

表 4-400 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-401 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
POST https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/members
{
  "users": [ {
    "user_id": "665b542eb4c14b53a15ead449f28b0ba",
    "role_id": 5
  } ]
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
```

```
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

import java.util.List;
import java.util.ArrayList;

public class BatchAddMembersV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        BatchAddMembersV4Request request = new BatchAddMembersV4Request();
        BatchAddMembersV4RequestBody body = new BatchAddMembersV4RequestBody();
        List<BatchAddMemberRequestV4> listbodyUsers = new ArrayList<>();
        listbodyUsers.add(
            new BatchAddMemberRequestV4()
                .withRoleId(5)
                .withUserId("665b542eb4c14b53a15ead449f28b0ba")
        );
        body.withUsers(listbodyUsers);
        request.withBody(body);
        try {
            BatchAddMembersV4Response response = client.batchAddMembersV4(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
```

```
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = __import__('os').getenv("CLOUD_SDK_AK")
sk = __import__('os').getenv("CLOUD_SDK_SK")

credentials = BasicCredentials(ak, sk) \

client = ProjectManClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = BatchAddMembersV4Request()
    listUsersbody = [
        BatchAddMemberRequestV4(
            role_id=5,
            user_id="665b542eb4c14b53a15ead449f28b0ba"
        )
    ]
    request.body = BatchAddMembersV4RequestBody(
        users=listUsersbody
    )
    response = client.batch_add_members_v4(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.BatchAddMembersV4Request{}
    roleIdUsers := int32(5)
    var listUsersbody = []model.BatchAddMemberRequestV4{
        {
            RoleId: &roleIdUsers,
            UserId: "665b542eb4c14b53a15ead449f28b0ba",
        },
    },
```



```
}
request.Body = &model.BatchAddMembersV4RequestBody{
    Users: listUsersbody,
}
response, err := client.BatchAddMembersV4(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.4.4 批量删除项目成员

功能介绍

批量删除项目成员

调用方法

请参见[如何调用API](#)。

URI

DELETE /v4/projects/{project_id}/members

表 4-402 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-403 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

表 4-404 请求 Body 参数

参数	是否必选	参数类型	描述
user_ids	是	Array of strings	用户id 最小长度：32 最大长度：32 数组长度：1 - 50

响应参数

状态码：400

表 4-405 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码：401

表 4-406 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
DELETE https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/members  
[ "665b542eb4c14b53a15ead449f28b0ba" ]
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;  
import com.huaweicloud.sdk.projectman.v4.*;  
import com.huaweicloud.sdk.projectman.v4.model.*;  
  
public class BatchDeleteMembersV4Solution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ProjectManClient client = ProjectManClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))  
            .build();  
        BatchDeleteMembersV4Request request = new BatchDeleteMembersV4Request();  
        request.withBatchDeleteMembersV4RequestBody("[665b542eb4c14b53a15ead449f28b0ba]");  
        request.withBody(listBatchDeleteMembersV4RequestBodyBatchDeleteMembersV4RequestBody);  
        try {  
            BatchDeleteMembersV4Response response = client.batchDeleteMembersV4(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getHttpStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = BatchDeleteMembersV4Request()
        request.body = listBatchDeleteMembersV4RequestBodyBatchDeleteMembersV4RequestBody
        response = client.batch_delete_members_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.BatchDeleteMembersV4Request{}
```

```
var batchDeleteMembersV4RequestBodyBatchDeleteMembersV4RequestBody interface{} =
"[665b542eb4c14b53a15ead449f28b0ba]"
request.Body = listBatchDeleteMembersV4RequestBodyBatchDeleteMembersV4RequestBody
response, err := client.BatchDeleteMembersV4(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.4.5 更新成员在项目中的角色

功能介绍

更新成员在项目中的角色

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/members/role

表 4-407 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-408 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

表 4-409 请求 Body 参数

参数	是否必选	参数类型	描述
role_id	是	Integer	成员角色, -1 项目创建者, 3 项目经理, 4 开发人员, 5 测试经理, 6 测试人员, 7 参与者, 8 浏览者, 9 运维经理
user_ids	是	Array of strings	用户id

响应参数

状态码：400

表 4-410 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码：401

表 4-411 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Post https://{endpoint}/v4/projects/e2da96a5d2c845e284f0ad47f8ca8cb1/members/role
{
  "user_ids" : [ "08777d423500d4b01fe5c008d04336ab" ],
  "role_id" : 7
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateMembesRoleV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        UpdateMembesRoleV4Request request = new UpdateMembesRoleV4Request();
        UpdateMembesRoleV4RequestBody body = new UpdateMembesRoleV4RequestBody();
        List<String> listbodyUserIds = new ArrayList<>();
        listbodyUserIds.add("08777d423500d4b01fe5c008d04336ab");
        body.withUserIds(listbodyUserIds);
        body.withRoleId(7);
        request.withBody(body);
        try {
            UpdateMembesRoleV4Response response = client.updateMembesRoleV4(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
        }
    }
}
```

```
e.printStackTrace();
System.out.println(e.getStatusCode());
System.out.println(e.getRequestId());
System.out.println(e.getErrorCode());
System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateMembesRoleV4Request()
        listUserIdsbody = [
            "08777d423500d4b01fe5c008d04336ab"
        ]
        request.body = UpdateMembesRoleV4RequestBody(
            user_ids=listUserIdsbody,
            role_id=7
        )
        response = client.update_membes_role_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```



```
ak := os.Getenv("CLOUD_SDK_AK")
sk := os.Getenv("CLOUD_SDK_SK")

auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := projectman.NewProjectManClient(
    projectman.ProjectManClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpdateMembesRoleV4Request{}
var listUserIdsbody = []string{
    "08777d423500d4b01fe5c008d04336ab",
}
request.Body = &model.UpdateMembesRoleV4RequestBody{
    UserIds: listUserIdsbody,
    RoleId: int32(7),
}
response, err := client.UpdateMembesRoleV4(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	ok
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.4.6 添加项目成员

功能介绍

添加项目成员,可以添加跨租户成员

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/member

表 4-412 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-413 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

表 4-414 请求 Body 参数

参数	是否必选	参数类型	描述
domain_name	否	String	租户名称（跨租户添加用户时，填写正确的租户名称，可将未授权的租户主动授权，将用户添加为项目成员）
domain_id	是	String	租户id
role_id	否	Integer	'用户在项目中的角色ID' 成员角色，-1 项目创建者, 3 项目经理, 4 开发人员, 5 测试经理, 6 测试人员, 7 参与者, 8 浏览者, 9 运维经理
user_id	是	String	用户32位uuid

响应参数

状态码：400

表 4-415 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-416 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

POST https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/member

```
{
  "user_id" : "665b542eb4c14b53a15ead449f28b0ba",
  "role_id" : 5,
  "domain_id" : "4e919d73499648e3b0292cd3cbef806a"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class AddMemberV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    }
}
```

```
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
AddMemberV4Request request = new AddMemberV4Request();
AddMemberRequestV4 body = new AddMemberRequestV4();
body.withUserId("665b542eb4c14b53a15ead449f28b0ba");
body.withRoleId(5);
body.withDomainId("4e919d73499648e3b0292cd3cbef806a");
request.withBody(body);
try {
    AddMemberV4Response response = client.addMemberV4(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = AddMemberV4Request()
        request.body = AddMemberRequestV4(
            user_id="665b542eb4c14b53a15ead449f28b0ba",
            role_id=5,
            domain_id="4e919d73499648e3b0292cd3cbef806a"
        )
        response = client.add_member_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
```

```
print(e.status_code)
print(e.request_id)
print(e.error_code)
print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.AddMemberV4Request{
        roleIdAddMemberRequestV4:= int32(5)
        request.Body = &model.AddMemberRequestV4{
            UserId: "665b542eb4c14b53a15ead449f28b0ba",
            RoleId: &roleIdAddMemberRequestV4,
            DomainId: "4e919d73499648e3b0292cd3cbef806a",
        }
    }
    response, err := client.AddMemberV4(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	No Content
400	Bad Request

状态码	描述
401	Unauthorized

错误码

请参见[错误码](#)。

4.4.7 AGC 调用 当前用户申请加入项目

功能介绍

AGC调用 当前用户申请加入项目, 申请的用户id写在header中

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/members/agc-join

表 4-417 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度: 32 最大长度: 32

请求参数

表 4-418 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度: 10 最大长度: 32768
Domain-Id	是	String	租户id 最小长度: 32 最大长度: 32

参数	是否必选	参数类型	描述
User-Id	是	String	用户id 最小长度：32 最大长度：32

响应参数

状态码：400

表 4-419 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码：401

表 4-420 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/members/agc-join
```

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/members/agc-join
```

响应示例

无

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.5 项目信息

4.5.1 获取项目详情

功能介绍

获取项目详情

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}

表 4-421 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id

请求参数

表 4-422 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码： 200

表 4-423 响应 Body 参数

参数	参数类型	描述
project	project object	项目信息

表 4-424 project

参数	参数类型	描述
project_num_id	Integer	项目numId
project_id	String	项目uuid
name	String	项目名称
created_on	Long	项目创建时间
updated_on	Long	项目更新时间
project_type	String	项目类型
archive	Integer	是否归档
enterprise_id	String	企业项目id
project_code	String	项目代号
creator	creator object	创建者信息

表 4-425 creator

参数	参数类型	描述
user_num_id	Integer	创建人num_id
user_id	String	创建人uuid
user_name	String	创建人姓名
domain_id	String	创建人租户id
domain_name	String	创建人租户名称
nick_name	String	创建人昵称

状态码： 400

表 4-426 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-427 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Get https://{endpoint}/v4/projects/0792b7d26cb74bd799cbae80063dc74a

响应示例

状态码： 200

OK

```
{
  "project": {
    "project_num_id": 3183801,
    "project_id": "41bd3eab86544863b8fc92ffcd35e3c8",
    "name": "demo_project",
    "created_on": 1576114296000,
    "updated_on": 1576114296000,
    "project_type": "scrum",
    "archive": 1,
    "enterprise_id": "041bd3eab86544863b8f92ffcd35e218",
    "project_code": "abc_123",
    "creator": {
      "user_num_id": 4097,
      "user_id": "c4bd39f0573a436589714bedbcca8536",
      "user_name": "modile_user",
      "domain_id": "2489d8cbba804db8bcd35384685ff3ac",
      "domain_name": "modile_user",
      "nick_name": "Tom.M.Riddle"
    }
  }
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ShowProjectInfoV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
```

```
environment variables and decrypted during use to ensure security.
// In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
String ak = System.getenv("CLOUD_SDK_AK");
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
ShowProjectInfoV4Request request = new ShowProjectInfoV4Request();
try {
    ShowProjectInfoV4Response response = client.showProjectInfoV4(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowProjectInfoV4Request()
        response = client.show_project_info_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowProjectInfoV4Request{}
    response, err := client.ShowProjectInfoV4(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.5.2 更新项目

功能介绍

更新项目

调用方法

请参见[如何调用API](#)。

URI

PUT /v4/projects/{project_id}

表 4-428 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-429 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-430 请求 Body 参数

参数	是否必选	参数类型	描述
description	否	String	项目描述
project_name	是	String	项目名 最小长度：1 最大长度：128

响应参数

状态码：400

表 4-431 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-432 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
PUT https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af
{
  "project_name": "ProjectUnionSDKTest_lj9",
  "description": "2222"
}
```

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class UpdateProjectV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
```

```
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
UpdateProjectV4Request request = new UpdateProjectV4Request();
UpdateProjectRequestV4 body = new UpdateProjectRequestV4();
body.withProjectName("ProjectUnionSDKTest_lj9");
body.withDescription("2222");
request.withBody(body);
try {
    UpdateProjectV4Response response = client.updateProjectV4(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateProjectV4Request()
        request.body = UpdateProjectRequestV4(
            project_name="ProjectUnionSDKTest_lj9",
            description="2222"
        )
        response = client.update_project_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
```

```
print(e.error_code)
print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateProjectV4Request{
        descriptionUpdateProjectRequestV4:= "2222"
    }
    request.Body = &model.UpdateProjectRequestV4{
        ProjectName: "ProjectUnionSDKTest_lj9",
        Description: &descriptionUpdateProjectRequestV4,
    }
    response, err := client.UpdateProjectV4(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.5.3 删除项目

功能介绍

删除项目

调用方法

请参见[如何调用API](#)。

URI

DELETE /v4/projects/{project_id}

表 4-433 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-434 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：400

表 4-435 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-436 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

DELETE https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af

响应示例

无

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class DeleteProjectV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);
```

```
ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
DeleteProjectV4Request request = new DeleteProjectV4Request();
try {
    DeleteProjectV4Response response = client.deleteProjectV4(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteProjectV4Request()
        response = client.delete_project_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteProjectV4Request{}
    response, err := client.DeleteProjectV4(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.5.4 查询项目列表

功能介绍

查询项目列表

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects

表 4-437 Query 参数

参数	是否必选	参数类型	描述
offset	是	Integer	偏移量 从0开始 最小值: 0 最大值: 10000 缺省值: 0
limit	是	Integer	条数 最小1条,最大1000 最小值: 1 最大值: 1000 缺省值: 10
search	否	String	模糊查询项目名称或描述,不支持通配符等高级查询
project_type	否	String	项目类型 scrum xboard
sort	否	String	排序条件 默认创建时间降序 (name created_on)(asc desc)
archive	否	String	是否归档 true已归档 false未归档
query_type	否	String	默认返回当前用户参与的项目列表, domain_projects租户下的所有项目列表, absent返回当前用户未参与的租户项目列表

请求参数

表 4-438 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取(响应消息头中X-Subject-Token的值)。 最小长度: 10 最大长度: 32768

响应参数

状态码： 200

表 4-439 响应 Body 参数

参数	参数类型	描述
projects	Array of projects objects	项目信息列表 数组长度： 0 - 100
total	Integer	项目总数 最小值： 0 最大值： 10000

表 4-440 projects

参数	参数类型	描述
project_num_id	Integer	项目numId
project_id	String	项目uuid
project_name	String	项目名称
description	String	项目描述
created_time	Long	项目创建时间
updated_time	Long	项目更新时间
project_type	String	项目类型
creator	creator object	创建者信息

表 4-441 creator

参数	参数类型	描述
user_num_id	Integer	创建人numId
user_id	String	创建人id
user_name	String	创建人姓名
domain_id	String	创建人租户id
domain_name	String	创建人租户名称
nick_name	String	创建人租户昵称

状态码： 400

表 4-442 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-443 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 403

表 4-444 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
GET https://{endpoint}/v4/projects?  
limit=10&offset=0&project_type=scrum&sort=&search=aa&archive=true&query_type=domain_projects
```

响应示例

状态码： 200

OK

```
{  
  "projects": [ {  
    "project_num_id": 3183801,  
    "project_id": "41bd3eab86544863b8fc92ffcd35e3c8",  
    "project_name": "demo_project",  
    "description": "demo",  
    "created_time": 1576114296000,  
    "updated_time": 1576114296000,  
    "project_type": "xboard",  
    "creator": {  
      "user_num_id": 4097,  
      "user_id": "c4bd39f0573a436589714bedbcca8536",  
    }  
  }  
]
```

```
"user_name" : "modile_user",
"domain_id" : "2489d8cbba804db8bcd35384685ff3ac",
"domain_name" : "modile_user",
"nick_name" : "Tom.M.Riddle"
}
}],
"total" : 10
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ListProjectsV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ListProjectsV4Request request = new ListProjectsV4Request();
        request.withOffset("<offset>");
        request.withLimit("<limit>");
        request.withSearch("<search>");
        request.withProjectType("<project_type>");
        request.withSort("<sort>");
        request.withArchive("<archive>");
        request.withQueryType("<query_type>");
        try {
            ListProjectsV4Response response = client.listProjectsV4(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```



```
}  
}
```

Python

```
# coding: utf-8  
  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkprojectman.v4 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = __import__('os').getenv("CLOUD_SDK_AK")  
    sk = __import__('os').getenv("CLOUD_SDK_SK")  
  
    credentials = BasicCredentials(ak, sk) \  
  
    client = ProjectManClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = ListProjectsV4Request()  
        request.offset = <offset>  
        request.limit = <limit>  
        request.search = "<search>"  
        request.project_type = "<project_type>"  
        request.sort = "<sort>"  
        request.archive = "<archive>"  
        request.query_type = "<query_type>"  
        response = client.list_projects_v4(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).
```

```
Build()

client := projectman.NewProjectManClient(
    projectman.ProjectManClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ListProjectsV4Request{}
request.Offset = int32(<offset>)
request.Limit = int32(<limit>)
searchRequest:= "<search>"
request.Search = &searchRequest
projectTypeRequest:= "<project_type>"
request.ProjectType = &projectTypeRequest
sortRequest:= "<sort>"
request.Sort = &sortRequest
archiveRequest:= "<archive>"
request.Archive = &archiveRequest
queryTypeRequest:= "<query_type>"
request.QueryType = &queryTypeRequest
response, err := client.ListProjectsV4(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized
403	Forbidden

错误码

请参见[错误码](#)。

4.5.5 获取租户没有加入的项目

功能介绍

获取租户没有加入的项目

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/domain/not-added

表 4-445 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	分页索引, 偏移量,offset是limit的整数倍, limit=10,offset=0,10,20... 最小值: 0 缺省值: 0
limit	否	Integer	每页显示的数量,每页最多显示100条 最小值: 1 最大值: 100 缺省值: 10

请求参数

表 4-446 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取(响应消息头中X-Subject-Token的值)。 最小长度: 10 最大长度: 32768

响应参数

状态码: 200

表 4-447 响应 Body 参数

参数	参数类型	描述
projects	Array of projects objects	项目信息列表 数组长度: 0 - 1000

参数	参数类型	描述
total	Integer	总数 最小值：0 最大值：10000

表 4-448 projects

参数	参数类型	描述
project_num_id	Integer	项目数字id
project_id	String	项目id 最小长度：32 最大长度：32
project_name	String	项目名 最小长度：1 最大长度：128
description	String	项目描述
created_time	Long	项目创建时间 最小长度：0 最大长度：13
updated_time	Long	项目更新时间 最小长度：0 最大长度：13
project_type	String	项目类型
creator	creator object	创建者信息

表 4-449 creator

参数	参数类型	描述
user_num_id	Integer	创建人numId
user_id	String	创建人id 最小长度：1 最大长度：32
user_name	String	创建人姓名

参数	参数类型	描述
domain_id	String	创建人租户id 最小长度：1 最大长度：32
domain_name	String	创建人租户名称
nick_name	String	创建人租户昵称

状态码：400

表 4-450 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码：401

表 4-451 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Get https://{endpoint}/v4/projects/domain/not-added
```

响应示例

状态码：200

OK

```
{
  "projects": [ {
    "project_num_id": 3056544,
    "project_id": "e2da96a5d2c845e284f0ad47f8ca8cb1",
    "project_name": "demo_project",
    "description": "这是一个demo",
    "created_time": "1571296951000",
    "updated_time": "1571296951000",
    "project_type": "scrum",
    "creator": {
      "user_num_id": 4097,
      "user_id": "c4bd39f0573a436589714bedbcca8536",
    }
  }
]
```

```
"user_name" : "modile_user",
"domain_id" : "2489d8cbba804db8bcd35384685ff3ac",
"domain_name" : "modile_user",
"nick_name" : "Tom.M.Riddle"
}
}],
"total" : 1
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ListDomainNotAddedProjectsV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ListDomainNotAddedProjectsV4Request request = new ListDomainNotAddedProjectsV4Request();
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        try {
            ListDomainNotAddedProjectsV4Response response = client.listDomainNotAddedProjectsV4(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListDomainNotAddedProjectsV4Request()
        request.offset = <offset>
        request.limit = <limit>
        response = client.list_domain_not_added_projects_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
```

```
request := &model.ListDomainNotAddedProjectsV4Request{}
offsetRequest:= int32(<offset>)
request.Offset = &offsetRequest
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
response, err := client.ListDomainNotAddedProjectsV4(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.5.6 创建项目

功能介绍

创建项目

调用方法

请参见[如何调用API](#)。

URI

POST /v4/project

请求参数

表 4-452 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

表 4-453 请求 Body 参数

参数	是否必选	参数类型	描述
project_name	是	String	项目名称 最小长度：1 最大长度：128
description	否	String	项目描述
source	否	String	项目来源
project_type	是	String	项目类型 scrum, xboard(看板项目), basic, phoenix(凤凰项目)
enterprise_id	否	String	项目要绑定的企业项目ID
template_id	否	Integer	用户创建的项目模板id

响应参数

状态码：200

表 4-454 响应 Body 参数

参数	参数类型	描述
project_num_id	Integer	项目数字id
project_id	String	项目id 最小长度：32 最大长度：32

参数	参数类型	描述
project_name	String	项目名 最小长度: 1 最大长度: 128
description	String	项目描述
project_type	String	项目类型
user_num_id	Integer	创建者的数字id

状态码: 400

表 4-455 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码: 401

表 4-456 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Post https://{endpoint}/v4/project
{
  "project_name": "TestDemo",
  "description": "modify testxxxx",
  "project_type": "scrum"
}
```

响应示例

状态码: 200

OK

```
{
  "project_num_id": 708168641,
  "project_name": "testdemo",
  "project_id": "2e76d6f0b2c5403796e36bf63c5d5561",
```

```
"user_num_id" : 700145037,  
"project_type" : "scrum",  
"description" : "this is demo"  
}
```

状态码： 400

Bad Request

```
{  
  "error_msg" : "param error",  
  "error_code" : "PM.00000001"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;  
import com.huaweicloud.sdk.projectman.v4.*;  
import com.huaweicloud.sdk.projectman.v4.model.*;  
  
public class CreateProjectV4Solution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ProjectManClient client = ProjectManClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))  
            .build();  
        CreateProjectV4Request request = new CreateProjectV4Request();  
        CreateProjectV4RequestBody body = new CreateProjectV4RequestBody();  
        body.withProjectType("scrum");  
        body.withDescription("modify testxxxx");  
        body.withProjectName("TestDemo");  
        request.withBody(body);  
        try {  
            CreateProjectV4Response response = client.createProjectV4(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getHttpStatusCode());  
            System.out.println(e.getRequestId());  
        }  
    }  
}
```

```
        System.out.println(e.getErrorCode());
        System.out.println(e.getErrorMsg());
    }
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateProjectV4Request()
        request.body = CreateProjectV4RequestBody(
            project_type="scrum",
            description="modify testxxxx",
            project_name="TestDemo"
        )
        response = client.create_project_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
```

```
WithSk(sk).
Build()

client := projectman.NewProjectManClient(
    projectman.ProjectManClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateProjectV4Request{}
descriptionCreateProjectV4RequestBody:= "modify testxxxx"
request.Body = &model.CreateProjectV4RequestBody{
    ProjectType: "scrum",
    Description: &descriptionCreateProjectV4RequestBody,
    ProjectName: "TestDemo",
}
response, err := client.CreateProjectV4(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.5.7 检查项目名称是否存在

功能介绍

检查项目名称是否存在

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/check-name

请求参数

表 4-457 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

表 4-458 请求 Body 参数

参数	是否必选	参数类型	描述
project_name	是	String	项目名 最小长度：1 最大长度：128

响应参数

状态码：200

表 4-459 响应 Body 参数

参数	参数类型	描述
exist	Boolean	是否存在相同的项目名称 true 存在，false 不存在

状态码：400

表 4-460 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码：401

表 4-461 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Post https://{endpoint}/v4/projects/check-name

```
{  
  "project_name": "testDemo"  
}
```

响应示例

状态码： 200

OK

```
{  
  "exist": true  
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;  
import com.huaweicloud.sdk.projectman.v4.*;  
import com.huaweicloud.sdk.projectman.v4.model.*;  
  
public class CheckProjectNameV4Solution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ProjectManClient client = ProjectManClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))  
            .build();  
    }  
}
```

```
CheckProjectNameV4Request request = new CheckProjectNameV4Request();
CheckProjectNameRequestV4 body = new CheckProjectNameRequestV4();
body.withProjectName("testDemo");
request.withBody(body);
try {
    CheckProjectNameV4Response response = client.checkProjectNameV4(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CheckProjectNameV4Request()
        request.body = CheckProjectNameRequestV4(
            project_name="testDemo"
        )
        response = client.check_project_name_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
```



```
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
  
    client := projectman.NewProjectManClient(  
        projectman.ProjectManClientBuilder().  
            WithRegion(region.ValueOf("<YOUR REGION>")).  
            WithCredential(auth).  
            Build())  
  
    request := &model.CheckProjectNameV4Request{  
        request.Body = &model.CheckProjectNameRequestV4{  
            ProjectName: "testDemo",  
        }  
    }  
    response, err := client.CheckProjectNameV4(request)  
    if err == nil {  
        fmt.Printf("%+v\n", response)  
    } else {  
        fmt.Println(err)  
    }  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.5.8 查询项目模板

功能介绍

查询项目模板

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/templates

表 4-462 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<p>参数解释: 项目的32位uuid, 项目唯一标识, 可以通过 ListProjectsV4接口查询项目列表</p> <p>约束限制: 正则表达式, <code>^[A-Za-z0-9]{32}\$</code></p> <p>取值范围: 字符串, 最小长度: 32, 最大长度: 32</p> <p>默认取值: 不涉及</p>

表 4-463 Query 参数

参数	是否必选	参数类型	描述
tracker_id	否	String	<p>参数解释: 工作项类型,</p> <p>约束限制: 任务/Task,3缺陷/Bug,5Epic,6Feature,7Story;</p> <p>取值范围: 最小值 2 最大值 7</p> <p>默认取值: 不涉及</p>

请求参数

表 4-464 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	<p>参数解释: 用户Token。通过调用IAM服务获取用户Token接口获取 (响应消息头中X-Subject-Token的值)。</p> <p>约束限制: 不涉及</p> <p>取值范围: 最小长度: 10, 最大长度: 32768</p> <p>默认取值: 不涉及</p>

响应参数

状态码： 200

表 4-465 响应 Body 参数

参数	参数类型	描述
templates	Array of templates objects	模板信息列表

表 4-466 templates

参数	参数类型	描述
id	Integer	模板id
project_id	Integer	项目的32位uuid，项目唯一标识，可以通过 ListProjectsV4接口查询项目列表
tracker_id	Integer	工作项类型id 枚举值： <ul style="list-style-type: none">• 2, 任务/Task• 3, 缺陷/Bug• 5, Epic• 6, Feature• 7, Story
description	String	工作项详情模板描述内容 最小长度：0 最大长度：65535
issue_field_config	String	模板配置

状态码： 400

表 4-467 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-468 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Get https://{endpoint}/v4/projects/fcd58d40cc4d46baaea02f2e6e75ee1f/templates?tracker_id=2
```

响应示例

状态码: 200

OK

```
{
  "templates": [ {
    "id": 1793674,
    "project_id": 30384422,
    "tracker_id": 2,
    "description": "",
    "issue_field_config": ""
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ListTemplatesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
```

```
        .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
        .build();
ListTemplatesRequest request = new ListTemplatesRequest();
request.withTrackerId("<tracker_id>");
try {
    ListTemplatesResponse response = client.listTemplates(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListTemplatesRequest()
        request.tracker_id = "<tracker_id>"
        response = client.list_templates(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)
```

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListTemplatesRequest{}
    trackerIdRequest := "<tracker_id>"
    request.TrackerId = &trackerIdRequest
    response, err := client.ListTemplates(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.6 项目指标

4.6.1 查询人均 bug

功能介绍

查询人均bug

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/bugs-per-developer/query

表 4-469 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id

请求参数

表 4-470 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-471 请求 Body 参数

参数	是否必选	参数类型	描述
metric_type	否	String	指标类型
sprint_id	否	String	迭代ID
dividend	否	dividend object	指标分子过滤条件
divisor	否	Object	指标分母过滤条件

表 4-472 dividend

参数	是否必选	参数类型	描述
custom_field1_6	否	String	自定义字段
custom_field1_7	否	String	自定义字段

参数	是否必选	参数类型	描述
custom_field18	否	String	自定义字段
custom_field19	否	String	自定义字段
custom_field20	否	String	自定义字段
custom_field21	否	String	自定义字段
custom_field22	否	String	自定义字段
custom_field23	否	String	自定义字段
custom_field24	否	String	自定义字段
custom_field25	否	String	自定义字段
custom_field26	否	String	自定义字段
custom_field27	否	String	自定义字段
custom_field28	否	String	自定义字段
custom_field29	否	String	自定义字段
custom_field30	否	String	自定义字段
custom_field31	否	String	自定义字段
custom_field32	否	String	自定义字段
custom_field33	否	String	自定义字段
custom_field34	否	String	自定义字段
custom_field35	否	String	自定义字段
custom_field36	否	String	自定义字段

参数	是否必选	参数类型	描述
custom_field37	否	String	自定义字段
custom_field38	否	String	自定义字段
custom_field39	否	String	自定义字段
custom_field40	否	String	自定义字段

响应参数

状态码： 200

表 4-473 响应 Body 参数

参数	参数类型	描述
dividend_value	String	指标分子数值
divisor_value	String	指标分母数值
metric_name	String	指标名称
metric_value	String	指标数值
project_id	String	项目ID
project_name	String	项目名称

状态码： 400

表 4-474 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-475 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Post https://{endpoint}/v1/eace9c0a00574de6878ab580daa9a317/bugs-per-developer/query

```
{
  "dividend": {
    "custom_field16": "自定义字段值"
  },
  "divisor": { },
  "metric_type": "issues_per_developer",
  "sprint_id": "8883443"
}
```

响应示例

状态码: 200

OK

```
{
  "dividend_value": 15,
  "divisor_value": 20,
  "metric_name": "completion_rate|bug_density|bugs_per_developer",
  "metric_value": 0.8945,
  "project_id": "eace9c0a00574de6878ab580daa9a317",
  "project_name": "demo"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ShowBugsPerDeveloperSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
```

```
String sk = System.getenv("CLOUD_SDK_SK");

ICredential auth = new BasicCredentials()
    .withAk(ak)
    .withSk(sk);

ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
ShowBugsPerDeveloperRequest request = new ShowBugsPerDeveloperRequest();
MetricRequest2 body = new MetricRequest2();
MetricRequest2Dividend dividendbody = new MetricRequest2Dividend();
dividendbody.withCustomField16("自定义字段值");
body.withDivisor(new Object());
body.withDividend(dividendbody);
body.withSprintId("8883443");
body.withMetricType("issues_per_developer");
request.withBody(body);
try {
    ShowBugsPerDeveloperResponse response = client.showBugsPerDeveloper(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowBugsPerDeveloperRequest()
        dividendbody = MetricRequest2Dividend(
            custom_field16="自定义字段值"
        )
        request.body = MetricRequest2(
            divisor={},
            dividend=dividendbody,
```

```
sprint_id="8883443",
metric_type="issues_per_developer"
)
response = client.show_bugs_per_developer(request)
print(response)
except exceptions.ClientRequestException as e:
print(e.status_code)
print(e.request_id)
print(e.error_code)
print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowBugsPerDeveloperRequest{
        customField16Dividend:= "自定义字段值"
        dividendbody := &model.MetricRequest2Dividend{
            CustomField16: &customField16Dividend,
        }
    }
    var divisorMetricRequest2 interface{} = make(map[string]string)
    sprintIdMetricRequest2:= "8883443"
    metricTypeMetricRequest2:= "issues_per_developer"
    request.Body = &model.MetricRequest2{
        Divisor: &divisorMetricRequest2,
        Dividend: dividendbody,
        SprintId: &sprintIdMetricRequest2,
        MetricType: &metricTypeMetricRequest2,
    }
    response, err := client.ShowBugsPerDeveloper(request)
    if err == nil {
        fmt.Printf("%v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.6.2 查询需求按时完成率

功能介绍

查询需求按时完成率

调用方法

请参见[如何调用API](#)。

URI

POST /v1/{project_id}/completion-rate/query

表 4-476 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id

请求参数

表 4-477 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-478 请求 Body 参数

参数	是否必选	参数类型	描述
date_range	否	String	统计周期
metric_type	否	String	指标类型
sprint_id	否	String	迭代ID
dividend	否	dividend object	指标分子过滤条件
divisor	否	Object	指标分母过滤条件

表 4-479 dividend

参数	是否必选	参数类型	描述
on_time	否	String	是否按时
custom_field1 6	否	String	自定义字段
custom_field1 7	否	String	自定义字段
custom_field1 8	否	String	自定义字段
custom_field1 9	否	String	自定义字段
custom_field2 0	否	String	自定义字段
custom_field2 1	否	String	自定义字段
custom_field2 2	否	String	自定义字段
custom_field2 3	否	String	自定义字段
custom_field2 4	否	String	自定义字段
custom_field2 5	否	String	自定义字段
custom_field2 6	否	String	自定义字段
custom_field2 7	否	String	自定义字段

参数	是否必选	参数类型	描述
custom_field28	否	String	自定义字段
custom_field29	否	String	自定义字段
custom_field30	否	String	自定义字段
custom_field31	否	String	自定义字段
custom_field32	否	String	自定义字段
custom_field33	否	String	自定义字段
custom_field34	否	String	自定义字段
custom_field35	否	String	自定义字段
custom_field36	否	String	自定义字段
custom_field37	否	String	自定义字段
custom_field38	否	String	自定义字段
custom_field39	否	String	自定义字段
custom_field40	否	String	自定义字段

响应参数

状态码： 200

表 4-480 响应 Body 参数

参数	参数类型	描述
dividend_value	String	指标分子数值
divisor_value	String	指标分母数值
metric_name	String	指标名称

参数	参数类型	描述
metric_value	String	指标数值
project_id	String	项目ID
project_name	String	项目名称

状态码： 400

表 4-481 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-482 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Post https://{endpoint}/v1/eace9c0a00574de6878ab580daa9a317/completion-rate/query

```
{
  "date_range": "1598457600000,1598544000000",
  "dividend": {
    "custom_field16": "自定义字段值",
    "on_time": "ontime"
  },
  "divisor": { },
  "metric_type": "on-time_completion_rate",
  "sprint_id": "8883443"
}
```

响应示例

状态码： 200

OK

```
{
  "dividend_value": 15,
  "divisor_value": 20,
  "metric_name": "completion_rate|bug_density|bugs_per_developer",
  "metric_value": 0.8945,
}
```



```
"project_id" : "eace9c0a00574de6878ab580daa9a317",  
"project_name" : "我的项目"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;  
import com.huaweicloud.sdk.projectman.v4.*;  
import com.huaweicloud.sdk.projectman.v4.model.*;  
  
public class ShowCompletionRateSolution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ProjectManClient client = ProjectManClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ShowCompletionRateRequest request = new ShowCompletionRateRequest();  
        MetricRequest3 body = new MetricRequest3();  
        MetricRequest3Dividend dividendbody = new MetricRequest3Dividend();  
        dividendbody.withOnTime("ontime")  
            .withCustomField16("自定义字段值");  
        body.withDivisor(new Object());  
        body.withDividend(dividendbody);  
        body.withSprintId("8883443");  
        body.withMetricType("on-time_completion_rate");  
        body.withDateRange("1598457600000,1598544000000");  
        request.withBody(body);  
        try {  
            ShowCompletionRateResponse response = client.showCompletionRate(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getHttpStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowCompletionRateRequest()
        dividendbody = MetricRequest3Dividend(
            on_time="ontime",
            custom_field16="自定义字段值"
        )
        request.body = MetricRequest3(
            divisor={},
            dividend=dividendbody,
            sprint_id="8883443",
            metric_type="on-time_completion_rate",
            date_range="1598457600000,1598544000000"
        )
        response = client.show_completion_rate(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
```

```
WithSk(sk).
Build()

client := projectman.NewProjectManClient(
    projectman.ProjectManClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.ShowCompletionRateRequest{
    onTimeDividend:= "ontime"
    customField16Dividend:= "自定义字段值"
    dividendbody := &model.MetricRequest3Dividend{
        OnTime: &onTimeDividend,
        CustomField16: &customField16Dividend,
    }
    var divisorMetricRequest3 interface{} = make(map[string]string)
    sprintIdMetricRequest3:= "8883443"
    metricTypeMetricRequest3:= "on-time_completion_rate"
    dateRangeMetricRequest3:= "1598457600000,1598544000000"
    request.Body = &model.MetricRequest3{
        Divisor: &divisorMetricRequest3,
        Dividend: dividendbody,
        SprintId: &sprintIdMetricRequest3,
        MetricType: &metricTypeMetricRequest3,
        DateRange: &dateRangeMetricRequest3,
    }
}
response, err := client.ShowCompletionRate(request)
if err == nil {
    fmt.Printf("%v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.6.3 查询缺陷密度

功能介绍

查询缺陷密度

调用方法

请参见[如何调用API](#)。

URI

POST /v2/{project_id}/bug-density/query

表 4-483 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id

请求参数

表 4-484 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-485 请求 Body 参数

参数	是否必选	参数类型	描述
date_range	否	String	统计周期
metric_type	否	String	指标类型
dividend	否	dividend object	指标分子过滤条件
divisor	否	Object	指标分母过滤条件

表 4-486 dividend

参数	是否必选	参数类型	描述
custom_fields	否	Array of custom_field s objects	自定义类型过滤条件

表 4-487 custom_fields

参数	是否必选	参数类型	描述
name	否	String	自定义字段名称
options	否	String	自定义字段取值，逗号分隔

响应参数

状态码： 200

表 4-488 响应 Body 参数

参数	参数类型	描述
dividend_value	String	指标分子数值
divisor_value	String	指标分母数值
metric_name	String	指标名称
metric_value	String	指标数值
project_id	String	项目ID
project_name	String	项目名称

状态码： 400

表 4-489 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-490 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Post https://{endpoint}/v2/eace9c0a00574de6878ab580daa9a317/bug-density/query
{
  "date_range": "1590940800000,1593446400000",
  "dividend": {
    "custom_fields": [ {
      "name": "自定义字段名称",
      "options": "自定义字段值"
    } ]
  },
  "divisor": { },
  "metric_type": "bug_density"
}
```

响应示例

状态码: 200

OK

```
{
  "dividend_value": 15,
  "divisor_value": 20,
  "metric_name": "completion_rate|bug_density|bugs_per_developer",
  "metric_value": 0.8945,
  "project_id": "eace9c0a00574de6878ab580daa9a317",
  "project_name": "demo"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

import java.util.List;
import java.util.ArrayList;

public class ShowBugDensityV2Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
```

```
        .withCredential(auth)
        .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
        .build();
ShowBugDensityV2Request request = new ShowBugDensityV2Request();
MetricRequestV2 body = new MetricRequestV2();
List<MetricRequestV2DividendCustomFields> listDividendCustomFields = new ArrayList<>();
listDividendCustomFields.add(
    new MetricRequestV2DividendCustomFields()
        .withName("自定义字段名称")
        .withOptions("自定义字段值")
);
MetricRequestV2Dividend dividendbody = new MetricRequestV2Dividend();
dividendbody.withCustomFields(listDividendCustomFields);
body.withDivisor(new Object());
body.withDividend(dividendbody);
body.withMetricType("bug_density");
body.withDateRange("1590940800000,1593446400000");
request.withBody(body);
try {
    ShowBugDensityV2Response response = client.showBugDensityV2(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowBugDensityV2Request()
        listCustomFieldsDividend = [
            MetricRequestV2DividendCustomFields(
                name="自定义字段名称",
                options="自定义字段值"
            )
        ]
        dividendbody = MetricRequestV2Dividend(
```

```
        custom_fields=listCustomFieldsDividend
    )
    request.body = MetricRequestV2(
        divisor={},
        dividend=dividendbody,
        metric_type="bug_density",
        date_range="1590940800000,1593446400000"
    )
    response = client.show_bug_density_v2(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowBugDensityV2Request{}
    nameCustomFields:= "自定义字段名称"
    optionsCustomFields:= "自定义字段值"
    var listCustomFieldsDividend = []model.MetricRequestV2DividendCustomFields{
        {
            Name: &nameCustomFields,
            Options: &optionsCustomFields,
        },
    }
    dividendbody := &model.MetricRequestV2Dividend{
        CustomFields: &listCustomFieldsDividend,
    }
    var divisorMetricRequestV2 interface{} = make(map[string]string)
    metricTypeMetricRequestV2:= "bug_density"
    dateRangeMetricRequestV2:= "1590940800000,1593446400000"
    request.Body = &model.MetricRequestV2{
        Divisor: &divisorMetricRequestV2,
        Dividend: dividendbody,
        MetricType: &metricTypeMetricRequestV2,
        DateRange: &dateRangeMetricRequestV2,
    }
}
```



```
response, err := client.ShowBugDensityV2(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.7 项目统计

4.7.1 获取 bug 统计信息

功能介绍

获取bug统计信息，按模块统计

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/bug-statistic

表 4-491 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	<p>参数解释: 项目的32位uuid, 项目唯一标识, 可以通过 ListProjectsV4接口查询项目列表</p> <p>约束限制: 正则表达式, <code>^[A-Za-z0-9]{32}\$</code></p> <p>取值范围: 字符串, 最小长度: 32, 最大长度: 32</p> <p>默认取值: 不涉及</p>

请求参数

表 4-492 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	<p>参数解释: 用户Token。通过调用IAM服务获取用户Token接口获取 (响应消息头中X-Subject-Token的值)。</p> <p>约束限制: 不涉及</p> <p>取值范围: 最小长度: 10, 最大长度: 32768</p> <p>默认取值: 不涉及</p>

响应参数

状态码: 200

表 4-493 响应 Body 参数

参数	参数类型	描述
bug_statistics	Array of BugStatisticResponseV4 objects	当前项目下bug统计信息

表 4-494 BugStatisticResponseV4

参数	参数类型	描述
critical_num	Integer	重要程度为关键的缺陷数，重要程度为关键对应id为10
defect_index	Double	DI值，不同重要程度的缺陷对应不同的分值
module	String	模块
normal_num	Integer	重要程度为一般的缺陷数，重要程度为一般对应id为12
serious_num	Integer	重要程度为严重的缺陷数，重要程度为严重对应id为11
tip_num	Integer	重要程度为提示的缺陷数，重要程度为提示对应id为13
total	Integer	当前项目下bug统计总数

状态码： 400

表 4-495 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-496 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/bug-statistic
```

响应示例

状态码： 200

OK

```
{
  "bug_statistics" : [ {
    "critical_num" : 0,
    "defect_index" : 1,
    "module" : "统计分数",
    "normal_num" : 1,
    "serious_num" : 0,
    "tip_num" : 0,
    "total" : 1
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ListProjectBugStaticsV4Solution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ListProjectBugStaticsV4Request request = new ListProjectBugStaticsV4Request();
        try {
            ListProjectBugStaticsV4Response response = client.listProjectBugStaticsV4(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListProjectBugStaticsV4Request()
        response = client.list_project_bug_statics_v4(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListProjectBugStaticsV4Request{}
    response, err := client.ListProjectBugStaticsV4(request)
```

```
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.7.2 获取需求统计信息

功能介绍

获取需求统计信息

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/demand-statistic

表 4-497 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

请求参数

表 4-498 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-499 响应 Body 参数

参数	参数类型	描述
demand_statistics	Array of DemandStatisticResponseV4 objects	需求统计

表 4-500 DemandStatisticResponseV4

参数	参数类型	描述
closed_num	Integer	已关闭数量
module	String	模块
new_num	Integer	新建的数量
process_num	Integer	开发中的数量
rejected_num	Integer	已拒绝数量
solved_num	Integer	已解决数量
test_num	Integer	测试中的数量
total	Integer	总数

状态码：400

表 4-501 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-502 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/demand-statistic
```

响应示例

状态码： 200

OK

```
{
  "demand_statistics": [{
    "closed_num": 1,
    "module": "计费",
    "new_num": 1,
    "process_num": 1,
    "rejected_num": 1,
    "solved_num": 1,
    "test_num": 1,
    "total": 1
  }]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;
```



```
public class ListProjectDemandStaticV4Solution {  
  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ProjectManClient client = ProjectManClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))  
            .build();  
        ListProjectDemandStaticV4Request request = new ListProjectDemandStaticV4Request();  
        try {  
            ListProjectDemandStaticV4Response response = client.listProjectDemandStaticV4(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getHttpStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

Python

```
# coding: utf-8  
  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkprojectman.v4 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = __import__('os').getenv("CLOUD_SDK_AK")  
    sk = __import__('os').getenv("CLOUD_SDK_SK")  
  
    credentials = BasicCredentials(ak, sk) \  
  
    client = ProjectManClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = ListProjectDemandStaticV4Request()  
        response = client.list_project_demand_static_v4(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)
```

```
print(e.request_id)
print(e.error_code)
print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListProjectDemandStaticV4Request{}
    response, err := client.ListProjectDemandStaticV4(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.7.3 获取项目概览

功能介绍

获取项目概览

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/summary

表 4-503 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	参数解释： 项目的32位uuid，项目唯一标识，可以通过ListProjectsV4接口查询项目列表 约束限制： 正则表达式， <code>^[A-Za-z0-9]{32}\$</code> 取值范围： 字符串，最小长度： 32 ，最大长度： 32 默认取值： 不涉及

请求参数

表 4-504 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	参数解释： 用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 约束限制： 不涉及 取值范围： 最小长度： 10 ，最大长度： 32768 默认取值： 不涉及

响应参数

状态码： 200

表 4-505 响应 Body 参数

参数	参数类型	描述
bug_statistics	Array of BugStatisticResponseV4 objects	bug统计列表
demand_statistics	Array of DemandStatisticResponseV4 objects	按模块统计列表
issue_completion_rates	Array of IssueCompletionRateResponseV4 objects	按工作项类型统计列表
project_id	String	项目的32位id，项目唯一标识

表 4-506 BugStatisticResponseV4

参数	参数类型	描述
critical_num	Integer	重要程度为关键的缺陷数，重要程度为关键对应id为10
defect_index	Double	DI值，不同重要程度的缺陷对应不同的分值
module	String	模块
normal_num	Integer	重要程度为一般的缺陷数，重要程度为一般对应id为12
serious_num	Integer	重要程度为严重的缺陷数，重要程度为严重对应id为11
tip_num	Integer	重要程度为提示的缺陷数，重要程度为提示对应id为13
total	Integer	当前项目下bug统计总数

表 4-507 DemandStatisticResponseV4

参数	参数类型	描述
closed_num	Integer	按模块统计已关闭数量
module	String	模块
new_num	Integer	按模块统计新建的数量
process_num	Integer	按模块统计开发中的数量
rejected_num	Integer	按模块统计已拒绝数量
solved_num	Integer	按模块统计已解决数量
test_num	Integer	按模块统计测试中的数量
total	Integer	按模块统计总数

表 4-508 IssueCompletionRateResponseV4

参数	参数类型	描述
issue_status	IssueStatusResponseV4 object	工作项状态维度的统计信息
tracker_id	Integer	工作项类型,2任务/Task,3缺陷/ Bug,5Epic,6Feature,7Story

表 4-509 IssueStatusResponseV4

参数	参数类型	描述
closed_num	Integer	当前项目中工作项已关闭数量
new_num	Integer	当前项目中工作项新建的数量
process_num	Integer	当前项目中工作项开发中的数量
rejected_num	Integer	当前项目中工作项已拒绝数量
solved_num	Integer	当前项目中工作项已解决数量
test_num	Integer	当前项目中工作项测试中的数量

状态码： 400

表 4-510 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-511 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/summary

GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/summary

响应示例

状态码： 200

OK

```
{
  "bug_statistics": [ {
    "critical_num": 0,
    "defect_index": 1,
    "module": "统计分数",
    "normal_num": 1,
    "serious_num": 0,
    "tip_num": 0,
    "total": 1
  } ],
  "demand_statistics": [ {
    "closed_num": 1,
    "module": "计费",
    "new_num": 1,
    "process_num": 1,
    "rejected_num": 1,
    "solved_num": 1,
    "test_num": 1,
    "total": 1
  } ],
  "issue_completion_rates": [ {
    "issue_status": {
      "closed_num": 1,
      "new_num": 1,
      "process_num": 1,
      "rejected_num": 1,
      "solved_num": 1,
      "test_num": 1
    }
  } ],
}
```

```
"tracker_id" : 0  
}],  
"project_id" : "526cefde62004de2b62b5e8dd2c2b3af"  
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.8 Scrum 项目的模块

4.8.1 创建项目的模块

功能介绍

查询项目的模块列表

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/module

表 4-512 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id

请求参数

表 4-513 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-514 请求 Body 参数

参数	是否必选	参数类型	描述
description	否	String	模块描述
module_name	是	String	模块名称
parent_module_id	否	Integer	父模块id
owner	是	UserRequest object	

表 4-515 UserRequest

参数	是否必选	参数类型	描述
user_id	是	String	用户32位字符串id

响应参数

状态码： 200

表 4-516 响应 Body 参数

参数	参数类型	描述
description	String	模块描述
module_name	String	模块名称
module_id	Integer	模块id
owner	ModuleOwner object	

表 4-517 ModuleOwner

参数	参数类型	描述
user_id	String	用户32位字符串id
user_num_id	Integer	用户数字id
user_name	String	用户名称
nick_name	String	用户昵称

状态码： 400

表 4-518 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-519 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Post https://{endpoint}/v4/projects/0792b7d26cb74bd799cbae80063dc74a/modules

```
{
  "description": "module demo",
  "module_name": "demo",
  "parent_module_id": 1,
  "owner": {
    "user_id": "09d75cde5180d4481f91c018e89cb8e2"
  }
}
```

响应示例

状态码： 200

OK

```
{
  "description": "module demo",
  "module_name": "demo",
}
```

```
"module_id" : 46691,
"owner" : {
  "user_id" : "09d75cde5180d4481f91c018e89cb8e2",
  "user_num_id" : 4091,
  "user_name" : "demo_user_name_06",
  "nick_name" : "模块demo"
}
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class CreateProjectModuleSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        CreateProjectModuleRequest request = new CreateProjectModuleRequest();
        CreateProjectModuleRequestBody body = new CreateProjectModuleRequestBody();
        UserRequest ownerbody = new UserRequest();
        ownerbody.withUserId("09d75cde5180d4481f91c018e89cb8e2");
        body.withOwner(ownerbody);
        body.withParentModuleId(1);
        body.withModuleName("demo");
        body.withDescription("module demo");
        request.withBody(body);
        try {
            CreateProjectModuleResponse response = client.createProjectModule(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

```
}  
}  
}
```

Python

```
# coding: utf-8  
  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkprojectman.v4 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = __import__('os').getenv("CLOUD_SDK_AK")  
    sk = __import__('os').getenv("CLOUD_SDK_SK")  
  
    credentials = BasicCredentials(ak, sk) \  
  
    client = ProjectManClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = CreateProjectModuleRequest()  
        ownerbody = UserRequest(  
            user_id="09d75cde5180d4481f91c018e89cb8e2"  
        )  
        request.body = CreateProjectModuleRequestBody(  
            owner=ownerbody,  
            parent_module_id=1,  
            module_name="demo",  
            description="module demo"  
        )  
        response = client.create_project_module(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")
```

```
auth := basic.NewCredentialsBuilder().
    WithAk(ak).
    WithSk(sk).
    Build()

client := projectman.NewProjectManClient(
    projectman.ProjectManClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.CreateProjectModuleRequest{
    ownerbody := &model.UserRequest{
        UserId: "09d75cde5180d4481f91c018e89cb8e2",
    }
}
parentModuleIdCreateProjectModuleRequestBody:= int32(1)
descriptionCreateProjectModuleRequestBody:= "module demo"
request.Body = &model.CreateProjectModuleRequestBody{
    Owner: ownerbody,
    ParentModuleId: &parentModuleIdCreateProjectModuleRequestBody,
    ModuleName: "demo",
    Description: &descriptionCreateProjectModuleRequestBody,
}
response, err := client.CreateProjectModule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.8.2 查询项目的模块列表

功能介绍

查询项目的模块列表

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/modules

表 4-520 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id

表 4-521 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	查询偏移量 缺省值: 0
limit	否	Integer	一次返回的数据,最小1,最大100 最小值: 1 最大值: 100 缺省值: 10

请求参数

表 4-522 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码: 200

表 4-523 响应 Body 参数

参数	参数类型	描述
total	Integer	模块总数
modules	Array of ProjectModule objects	模块列表

表 4-524 ProjectModule

参数	参数类型	描述
module_id	Integer	模块id
module_name	String	模块名称
owner	ModuleOwner object	
depth	Integer	模块层级
is_parent	Boolean	是否是父级, true 父模块, false 非父模块
children	Array of ProjectChildModule objects	子模块信息

表 4-525 ProjectChildModule

参数	参数类型	描述
module_id	Integer	模块id
module_name	String	模块名称
owner	ModuleOwner object	
depth	Integer	模块层级
is_parent	Boolean	是否是父级, true 父模块, false 非父模块
parent_module_id	Integer	父模块id

表 4-526 ModuleOwner

参数	参数类型	描述
user_id	String	用户32位字符串id
user_num_id	Integer	用户数字id
user_name	String	用户名称
nick_name	String	用户昵称

状态码: 400

表 4-527 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-528 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Get https://{endpoint}/v4/projects/0792b7d26cb74bd799cbae80063dc74a/modules

响应示例

状态码： 200

OK

```
{
  "total": 1,
  "modules": [ {
    "module_id": 1,
    "module_name": "demo",
    "owner": {
      "user_id": "09d75cde5180d4481f91c018e89cb8e2",
      "user_num_id": 4091,
      "user_name": "demo_user_name_06",
      "nick_name": "demo"
    },
    "depth": 1,
    "is_parent": true,
    "children": [ {
      "module_id": 1,
      "module_name": "demo",
      "owner": {
        "user_id": "09d75cde5180d4481f91c018e89cb8e2",
        "user_num_id": 4091,
        "user_name": "demo_user_name_06",
        "nick_name": "demo"
      },
      "depth": 1,
      "is_parent": true,
      "parent_module_id": 1
    } ]
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ListProjectModulesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ListProjectModulesRequest request = new ListProjectModulesRequest();
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        try {
            ListProjectModulesResponse response = client.listProjectModules(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
```


risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.

In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment

```
ak = __import__('os').getenv("CLOUD_SDK_AK")
sk = __import__('os').getenv("CLOUD_SDK_SK")

credentials = BasicCredentials(ak, sk) \

client = ProjectManClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ListProjectModulesRequest()
    request.offset = <offset>
    request.limit = <limit>
    response = client.list_project_modules(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListProjectModulesRequest{}
    offsetRequest:= int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest:= int32(<limit>)
    request.Limit = &limitRequest
    response, err := client.ListProjectModules(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

```
}  
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.8.3 更新项目的模块

功能介绍

更新项目的模块

调用方法

请参见[如何调用API](#)。

URI

PUT /v4/projects/{project_id}/modules/{module_id}

表 4-529 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32
module_id	是	Integer	模块id 最小长度：32 最大长度：32

请求参数

表 4-530 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-531 请求 Body 参数

参数	是否必选	参数类型	描述
description	否	String	模块描述
module_name	是	String	模块名称
owner	是	UserRequest object	

表 4-532 UserRequest

参数	是否必选	参数类型	描述
user_id	是	String	用户32位字符串id

响应参数

状态码：200

表 4-533 响应 Body 参数

参数	参数类型	描述
description	String	模块描述
module_name	String	模块名称
module_id	Integer	模块id
owner	ModuleOwner object	

表 4-534 ModuleOwner

参数	参数类型	描述
user_id	String	用户32位字符串id
user_num_id	Integer	用户数字id
user_name	String	用户名称
nick_name	String	用户昵称

状态码： 400

表 4-535 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-536 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
PUT https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/modules/662086
{
  "description": "module demo",
  "module_name": "demo",
  "owner": {
    "user_id": "09d75cde5180d4481f91c018e89cb8e2"
  }
}
```

响应示例

状态码： 200

OK

```
{
  "description": "module name",
  "module_name": "demo",
  "module_id": 46691,
}
```

```
"owner" : {  
  "user_id" : "09d75cde5180d4481f91c018e89cb8e2",  
  "user_num_id" : 4091,  
  "user_name" : "demo_user_name_06",  
  "nick_name" : "demo"  
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;  
  
import com.huaweicloud.sdk.core.auth.ICredential;  
import com.huaweicloud.sdk.core.auth.BasicCredentials;  
import com.huaweicloud.sdk.core.exception.ConnectionException;  
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;  
import com.huaweicloud.sdk.core.exception.ServiceResponseException;  
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;  
import com.huaweicloud.sdk.projectman.v4.*;  
import com.huaweicloud.sdk.projectman.v4.model.*;  
  
public class UpdateProjectModuleSolution {  
    public static void main(String[] args) {  
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great  
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or  
        // environment variables and decrypted during use to ensure security.  
        // In this example, AK and SK are stored in environment variables for authentication. Before running  
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
        String ak = System.getenv("CLOUD_SDK_AK");  
        String sk = System.getenv("CLOUD_SDK_SK");  
  
        ICredential auth = new BasicCredentials()  
            .withAk(ak)  
            .withSk(sk);  
  
        ProjectManClient client = ProjectManClient.newBuilder()  
            .withCredential(auth)  
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))  
            .build();  
        UpdateProjectModuleRequest request = new UpdateProjectModuleRequest();  
        UpdateProjectModuleRequestBody body = new UpdateProjectModuleRequestBody();  
        UserRequest ownerbody = new UserRequest();  
        ownerbody.withUserId("09d75cde5180d4481f91c018e89cb8e2");  
        body.withOwner(ownerbody);  
        body.withModuleName("demo");  
        body.withDescription("module demo");  
        request.withBody(body);  
        try {  
            UpdateProjectModuleResponse response = client.updateProjectModule(request);  
            System.out.println(response.toString());  
        } catch (ConnectionException e) {  
            e.printStackTrace();  
        } catch (RequestTimeoutException e) {  
            e.printStackTrace();  
        } catch (ServiceResponseException e) {  
            e.printStackTrace();  
            System.out.println(e.getStatusCode());  
            System.out.println(e.getRequestId());  
            System.out.println(e.getErrorCode());  
            System.out.println(e.getErrorMsg());  
        }  
    }  
}
```

```
}  
}
```

Python

```
# coding: utf-8  
  
from huaweicloudsdkcore.auth.credentials import BasicCredentials  
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion  
from huaweicloudsdkcore.exceptions import exceptions  
from huaweicloudsdkprojectman.v4 import *  
  
if __name__ == "__main__":  
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    # variables and decrypted during use to ensure security.  
    # In this example, AK and SK are stored in environment variables for authentication. Before running this  
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak = __import__('os').getenv("CLOUD_SDK_AK")  
    sk = __import__('os').getenv("CLOUD_SDK_SK")  
  
    credentials = BasicCredentials(ak, sk) \  
  
    client = ProjectManClient.new_builder() \  
        .with_credentials(credentials) \  
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \  
        .build()  
  
    try:  
        request = UpdateProjectModuleRequest()  
        ownerbody = UserRequest(  
            user_id="09d75cde5180d4481f91c018e89cb8e2"  
        )  
        request.body = UpdateProjectModuleRequestBody(  
            owner=ownerbody,  
            module_name="demo",  
            description="module demo"  
        )  
        response = client.update_project_module(request)  
        print(response)  
    except exceptions.ClientRequestException as e:  
        print(e.status_code)  
        print(e.request_id)  
        print(e.error_code)  
        print(e.error_msg)
```

Go

```
package main  
  
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"  
)  
  
func main() {  
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security  
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment  
    // variables and decrypted during use to ensure security.  
    // In this example, AK and SK are stored in environment variables for authentication. Before running this  
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment  
    ak := os.Getenv("CLOUD_SDK_AK")  
    sk := os.Getenv("CLOUD_SDK_SK")  
  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).
```

```
WithSk(sk).
Build()

client := projectman.NewProjectManClient(
    projectman.ProjectManClientBuilder().
        WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

request := &model.UpdateProjectModuleRequest{}
ownerbody := &model.UserRequest{
    UserId: "09d75cde5180d4481f91c018e89cb8e2",
}
descriptionUpdateProjectModuleRequestBody:= "module demo"
request.Body = &model.UpdateProjectModuleRequestBody{
    Owner: ownerbody,
    ModuleName: "demo",
    Description: &descriptionUpdateProjectModuleRequestBody,
}
response, err := client.UpdateProjectModule(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.8.4 删除项目的模块

功能介绍

删除项目的模块

调用方法

请参见[如何调用API](#)。

URI

DELETE /v4/projects/{project_id}/modules/{module_id}

表 4-537 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32
module_id	是	Integer	模块id 最小长度：32 最大长度：32

请求参数

表 4-538 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：400

表 4-539 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码：401

表 4-540 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
DELETE https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/modules/6354
```

```
DELETE https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/modules/6354
```

响应示例

无

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.9 Scrum 项目的领域

4.9.1 创建项目的领域

功能介绍

查询项目的领域列表

调用方法

请参见[如何调用API](#)。

URI

POST /v4/projects/{project_id}/domain

表 4-541 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id

请求参数

表 4-542 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-543 请求 Body 参数

参数	是否必选	参数类型	描述
domain_name	是	String	领域名称

响应参数

状态码： 200

表 4-544 响应 Body 参数

参数	参数类型	描述
domain_name	String	领域名称
domain_id	String	领域id

状态码： 400

表 4-545 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-546 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
Post https://{endpoint}/v4/projects/0792b7d26cb74bd799cbae80063dc74a/domains
{
  "domain_name" : "demo"
}
```

响应示例

状态码： 200

OK

```
{
  "domain_name" : "demo",
  "domain_id" : "123cefde62004de2b62b5e8dd2c2b3af"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class CreateProjectDomainSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);
```

```
ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
CreateProjectDomainRequest request = new CreateProjectDomainRequest();
CreateProjectDomainRequestBody body = new CreateProjectDomainRequestBody();
body.withDomainName("demo");
request.withBody(body);
try {
    CreateProjectDomainResponse response = client.createProjectDomain(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getHttpStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateProjectDomainRequest()
        request.body = CreateProjectDomainRequestBody(
            domain_name="demo"
        )
        response = client.create_project_domain(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateProjectDomainRequest{}
    request.Body = &model.CreateProjectDomainRequestBody{
        DomainName: "demo",
    }
    response, err := client.CreateProjectDomain(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.9.2 查询项目的领域列表

功能介绍

查询项目的领域列表

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/domains

表 4-547 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id

表 4-548 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	查询偏移量 缺省值: 0
limit	否	Integer	一次返回的数据,最小1,最大100 最小值: 1 最大值: 100 缺省值: 10

请求参数

表 4-549 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码: 200

表 4-550 响应 Body 参数

参数	参数类型	描述
total	Integer	领域总数
domains	Array of CreateProjectDomainResponseBody objects	领域列表

表 4-551 CreateProjectDomainResponseBody

参数	参数类型	描述
domain_name	String	领域名称
domain_id	String	领域id

状态码： 400

表 4-552 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-553 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Get <https://{endpoint}/v4/projects/0792b7d26cb74bd799cbae80063dc74a/domains>

响应示例

状态码： 200

OK

```
{
  "total": 1,
  "domains": [ {
    "domain_name": "demo",
    "domain_id": "123cefde62004de2b62b5e8dd2c2b3af"
  } ]
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ListProjectDomainsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ListProjectDomainsRequest request = new ListProjectDomainsRequest();
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        try {
            ListProjectDomainsResponse response = client.listProjectDomains(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```


Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListProjectDomainsRequest()
        request.offset = <offset>
        request.limit = <limit>
        response = client.list_project_domains(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    // risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    // variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    // example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
```

```
request := &model.ListProjectDomainsRequest{}
offsetRequest:= int32(<offset>)
request.Offset = &offsetRequest
limitRequest:= int32(<limit>)
request.Limit = &limitRequest
response, err := client.ListProjectDomains(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.9.3 更新项目的领域

功能介绍

更新项目的领域

调用方法

请参见[如何调用API](#)。

URI

PUT /v4/projects/{project_id}/domains/{domain_id}

表 4-554 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

参数	是否必选	参数类型	描述
domain_id	是	String	领域id 最小长度：32 最大长度：32

请求参数

表 4-555 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

表 4-556 请求 Body 参数

参数	是否必选	参数类型	描述
domain_name	是	String	领域名称

响应参数

状态码：200

表 4-557 响应 Body 参数

参数	参数类型	描述
domain_name	String	领域名称
domain_id	String	领域id

状态码：400

表 4-558 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-559 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
PUT https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/domains/662086
{
  "domain_name" : "test"
}
```

响应示例

状态码： 200

OK

```
{
  "domain_name" : "领域demo",
  "domain_id" : "123cefde62004de2b62b5e8dd2c2b3af"
}
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class UpdateProjectDomainSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);
```

```
ProjectManClient client = ProjectManClient.newBuilder()
    .withCredential(auth)
    .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
    .build();
UpdateProjectDomainRequest request = new UpdateProjectDomainRequest();
CreateProjectDomainRequestBody body = new CreateProjectDomainRequestBody();
body.withDomainName("test");
request.withBody(body);
try {
    UpdateProjectDomainResponse response = client.updateProjectDomain(request);
    System.out.println(response.toString());
} catch (ConnectionException e) {
    e.printStackTrace();
} catch (RequestTimeoutException e) {
    e.printStackTrace();
} catch (ServiceResponseException e) {
    e.printStackTrace();
    System.out.println(e.getStatusCode());
    System.out.println(e.getRequestId());
    System.out.println(e.getErrorCode());
    System.out.println(e.getErrorMsg());
}
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    # risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    # variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
    # example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = __import__('os').getenv("CLOUD_SDK_AK")
    sk = __import__('os').getenv("CLOUD_SDK_SK")

    credentials = BasicCredentials(ak, sk) \

    client = ProjectManClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateProjectDomainRequest()
        request.body = CreateProjectDomainRequestBody(
            domain_name="test"
        )
        response = client.update_project_domain(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
```

```
"github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
"github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
    risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
    variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
    example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateProjectDomainRequest{}
    request.Body = &model.CreateProjectDomainRequestBody{
        DomainName: "test",
    }
    response, err := client.UpdateProjectDomain(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.9.4 取消领域与项目的关联关系

功能介绍

取消领域与项目的关联关系

调用方法

请参见[如何调用API](#)。

URI

DELETE /v4/projects/{project_id}/domains/{domain_id}

表 4-560 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32
domain_id	是	String	领域id 最小长度：32 最大长度：32

请求参数

表 4-561 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：400

表 4-562 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-563 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
DELETE https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/domains/6354
DELETE https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/domains/6354
```

响应示例

无

状态码

状态码	描述
204	No Content
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.10 看板项目的工作项

4.10.1 查询看板项目下工作项的状态历史记录

功能介绍

分页查询看板项目下工作项的状态历史记录

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/work-items/status-records

表 4-564 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

表 4-565 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量 从0开始 最小值：0 缺省值：0
limit	否	Integer	每页数量 最小1,最大100 最小值：1 最大值：100 缺省值：100

请求参数

表 4-566 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-567 响应 Body 参数

参数	参数类型	描述
records	Array of WorkitemStatusRecords objects	操作历史
total	Integer	总数

表 4-568 WorkitemStatusRecords

参数	参数类型	描述
work_item_record_id	String	工作项的记录id，一个工作项对应一条记录 最小长度：32 最大长度：32
work_item_id	String	工作项id
project_id	String	devcloud项目的32位id 最小长度：32 最大长度：32
work_item_statuses	Array of WorkitemStatus objects	操作历史

表 4-569 WorkitemStatus

参数	参数类型	描述
id	String	工作项状态变更记录的id,每次变更产生一条新的记录id 最小长度: 1 最大长度: 32
status	status object	工作项的状态

表 4-570 status

参数	参数类型	描述
id	String	工作项的状态id 最小长度: 1 最大长度: 32
name	String	状态名称 最小长度: 1 最大长度: 60
type	String	工作项状态的类型, BACKLOG("初始化"), READY("待启动"), IN_PROGRESS("进行中"), COMPLETE("已完成"), DONE("已结束"),
description	String	工作项状态的描述
parent_status_id	String	工作项状态的父状态id 最小长度: 1 最大长度: 32

状态码: 400

表 4-571 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码: 401

表 4-572 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/work-items/status-records
```

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/work-items/status-records
```

响应示例

状态码: 200

OK

```
{
  "records": [ {
    "work_item_record_id": "7401906a85de466c81fd90e22824eb19",
    "work_item_id": "4633454879781163008",
    "project_id": "526cefde62004de2b62b5e8dd2c2b3af",
    "work_item_statuses": [ {
      "id": "7401906a85de466c81fd90e22824eb19",
      "status": {
        "id": "d25fe91f7f6b489db5f804e66ac3b360",
        "name": "7401906a85de466c81fd90e22824eb19",
        "type": "BACKLOG",
        "description": "demo",
        "parent_status_id": "7401906a85de466c81fd90e22824aaaa"
      }
    }
  ]
},
  "total": 1
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.10.2 查询看板项目下的工作项

功能介绍

查询看板项目下的工作项

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/work-items

表 4-573 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

表 4-574 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	偏移量 从0开始 最小值：0 缺省值：0
limit	否	Integer	每页数量 最小1,最大100 最小值：1 最大值：100 缺省值：100
created_time_ interval	否	String	创建工作项的时间(查询的起始 时间,查询的结束时间)

请求参数

表 4-575 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-576 响应 Body 参数

参数	参数类型	描述
work_items	Array of Workitems objects	工作项
total	Integer	总数

表 4-577 Workitems

参数	参数类型	描述
id	String	工作项id
description	String	工作项描述
actual_work_hours	Double	实际工时
assigned_user	WorkitemUser object	
author	WorkitemUser object	
begin_time	String	工作项开始时间
created_time	String	创建时间
tags	Array of tags objects	标签

参数	参数类型	描述
developer	WorkitemUser object	
assigned_cc_user	Array of WorkitemUser objects	抄送人
discover_version	String	发现问题的版本
end_time	String	工作项结束时间
done_ratio	String	工作项进度值
expected_work_hours	Double	预计工时
order	String	顺序
parent_work_item_id	String	父工作项的id
release_version	String	发布的版本
story_point	String	故事点
domain	domain object	id, 领域 14, '性能', 15, '功能', 16, '可靠性' 17, '网络安全' 18, '可维护性' 19, '其他DFX' 20, '可用性' 其他
iteration	iteration object	迭代
module	module object	模块
priority	String	工作项优先级
severity	String	严重的程度 "提示", "一般", "严重", "致命"
status	status object	工作项状态
subject	String	工作项标题
updated_time	String	更新时间
sequence	String	工作项编号
important	String	重要程度 "关键", "重要", "一般", "提示"
custom_fields	Array of WorkitemCustomField objects	用户自定义字段

表 4-578 tags

参数	参数类型	描述
id	String	标签id
name	String	标签名

表 4-579 WorkitemUser

参数	参数类型	描述
id	String	用户32位uuid
name	String	用户名
nick_name	String	昵称

表 4-580 domain

参数	参数类型	描述
id	String	领域id
name	String	领域

表 4-581 iteration

参数	参数类型	描述
id	String	迭代id
name	String	迭代名

表 4-582 module

参数	参数类型	描述
id	String	模块id
name	String	模块

表 4-583 status

参数	参数类型	描述
id	String	状态id

参数	参数类型	描述
name	String	状态名称

表 4-584 WorkitemCustomField

参数	参数类型	描述
field_id	String	自定义字段id
field_name	String	自定义字段名称
field_type	String	自定义字段类型, "Date","Number","DateTime", "MultiLineText","SingleLineText", "Select", "Checkbox"
field_option_source	String	自定义字段的选项源,CUSTOM, USER, DOMAIN, ITERATION, MODULE, TAG
value	String	自定义字段值, (field_type为Date,Number,DateTime时,field_option_source为空, value值是数字的字符串), (field_type为MultiLineText,SingleLineText时,field_option_source为空, value值是文本字符串), (field_type为Select ,field_option_source为CUSTOM时, value值是文本字符串) (field_type为Select ,field_option_source为用户, DOMAIN, ITERATION, MODULE, TAG时, value值是Json格式{}), (field_type为Checkbox ,field_option_source为CUSTOM时, value值是字符串数组["aaa"]), (field_type为"Checkbox" ,field_option_source为用户, DOMAIN, ITERATION, MODULE, TAG时, value值是Json的数组[{}])

状态码: 400

表 4-585 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码: 401

表 4-586 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/work-items

GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/work-items

响应示例

状态码： 200

OK

```
{
  "work_items": [ {
    "id": "4633454879781163008",
    "description": "demo",
    "actual_work_hours": 10,
    "assigned_user": {
      "id": "665b542eb4c14b53a15ead449f28b0ba",
      "name": "demo_user_name",
      "nick_name": "demo"
    },
    "author": {
      "id": "665b542eb4c14b53a15ead449f28b0ba",
      "name": "demo_user_name",
      "nick_name": "demo"
    },
    "begin_time": "1590940800000",
    "created_time": "1590940800000",
    "tags": [ {
      "id": "456",
      "name": "tagdemo"
    } ],
    "developer": {
      "id": "665b542eb4c14b53a15ead449f28b0ba",
      "name": "demo_user_name",
      "nick_name": "demo"
    },
    "assigned_cc_user": [ {
      "id": "665b542eb4c14b53a15ead449f28b0ba",
      "name": "demo_user_name",
      "nick_name": "demo"
    } ],
    "discover_version": "ProjectMan 8.0.0",
    "end_time": "1590940800000",
    "done_ratio": "10%",
    "expected_work_hours": 9,
    "order": "1",
    "parent_work_item_id": "4633454879781163001",
    "release_version": "ProjectMan 8.0.0",
    "story_point": "0",
    "domain": {
      "id": "14",
      "name": "性能"
    },
    "iteration": {
      "id": "456",
```

```
{
  "name": "迭代1"
},
"module": {
  "id": "456",
  "name": "Project"
},
"priority": "低",
"severity": "一般",
"status": {
  "id": "d25fe91f7f6b489db5f804e66ac3b360",
  "name": "新建"
},
"subject": "demosubject",
"updated_time": "1595832054113",
"sequence": "5500756",
"important": "提示",
"custom_fields": [ {
  "field_id": "d25fe91f7f6b489db5f804e66a0",
  "field_name": "customField",
  "field_type": "Select",
  "field_option_source": "USER",
  "value": "{\"id\":\"d25fe91f7f6b489db5f804e66a0\",\"name\":\"devcloud_test001\",\"nick_name\":\"demo\"}"
} ]
}],
"total": 1
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.10.3 查询看板项目的工作项流转配置

功能介绍

查询看板项目的工作项流转配置

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/work-items/workflow/config

表 4-587 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id 最小长度：32 最大长度：32

表 4-588 Query 参数

参数	是否必选	参数类型	描述
board_id	是	String	看板id

请求参数

表 4-589 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。 最小长度：10 最大长度：32768

响应参数

状态码：200

表 4-590 响应 Body 参数

参数	参数类型	描述
workflows	Array of WorkItemStatusFlowVo objects	流转数据

表 4-591 WorkItemStatusFlowVo

参数	参数类型	描述
parent_name	String	父状态的名称

参数	参数类型	描述
parent_type	String	父状态的类型
status_id	String	状态id
name	String	状态名称
status_type	String	状态类型
direct_to	Array of StatusFlowDirectToVo objects	流转到的数据
assign_to	String	处理人的uuid
comment	String	评论内容
required_assignment	Boolean	处理人是否必填
required_notes	Boolean	评论是否必填
field_type	Boolean	是否是字段值, true 处理人的信息是字段值, false 处理人的值是用户的信息,固定值
parent_id	String	父状态的uuid

表 4-592 StatusFlowDirectToVo

参数	参数类型	描述
parent_name	String	父状态的名称
parent_type	String	父状态的类型
status_id	String	状态id
name	String	状态名称
status_type	String	状态类型
enabled	Boolean	是否已开启状态流转, true: 开启, false 没开启
parent_id	String	父状态的uuid

状态码: 400

表 4-593 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-594 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/work-items/workflow/config?board_id=5aef890d5a2d45b2b22f3e307e785715
```

```
GET https://{endpoint}/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/work-items/workflow/config?board_id=5aef890d5a2d45b2b22f3e307e785715
```

响应示例

状态码： 200

OK

```
{
  "workflows" : [ {
    "parent_name" : "进行中",
    "parent_type" : "IN_PROGRESS",
    "status_id" : "d25fe91f7f6b489db5f804e66ac3b360",
    "name" : "研发",
    "status_type" : "IN_PROGRESS",
    "direct_to" : [ {
      "parent_name" : "进行中",
      "parent_type" : "IN_PROGRESS",
      "status_id" : "d25fe91f7f6b489db5f804e66ac3b360",
      "name" : "研发",
      "status_type" : "IN_PROGRESS",
      "enabled" : false,
      "parent_id" : "beda5e42974c42a09e142faa1dc0fc98"
    } ],
    "assign_to" : "a360371833bf4c558f796fd707b44daf",
    "comment" : "A transfer to B",
    "required_assign" : false,
    "required_notes" : false,
    "field_type" : false,
    "parent_id" : "beda5e42974c42a09e142faa1dc0fc98"
  } ]
}
```

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

4.11 Scrum 项目的状态

4.11.1 查询项目的状态列表

功能介绍

查询项目的状态列表

调用方法

请参见[如何调用API](#)。

URI

GET /v4/projects/{project_id}/statuses

表 4-595 路径参数

参数	是否必选	参数类型	描述
project_id	是	String	devcloud项目的32位id

表 4-596 Query 参数

参数	是否必选	参数类型	描述
offset	否	Integer	查询偏移量 缺省值: 0
limit	否	Integer	一次返回的数据,最小1,最大100 最小值: 1 最大值: 100 缺省值: 10

参数	是否必选	参数类型	描述
tracker_id	否	Integer	自定义字段支持的工作项类型 2 任务/Task,3缺陷/Bug,5Epic,6Feature,7Story

请求参数

表 4-597 请求 Header 参数

参数	是否必选	参数类型	描述
X-Auth-Token	是	String	用户Token。通过调用IAM服务获取用户Token接口获取（响应消息头中X-Subject-Token的值）。

响应参数

状态码： 200

表 4-598 响应 Body 参数

参数	参数类型	描述
total	Integer	状态总数
issue_statuses	Array of IssueStatus objects	状态列表

表 4-599 IssueStatus

参数	参数类型	描述
id	String	状态uuid
status_id	Integer	状态数字id
name	String	状态名称
tracker_ids	Array of integers	关联的工作项类型列表
status_attribute	StatusAttribute object	状态属性类型 开始态,进行态,结束态

表 4-600 StatusAttribute

参数	参数类型	描述
id	Integer	状态属性id
name	String	状态属性名称

状态码： 400

表 4-601 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

状态码： 401

表 4-602 响应 Body 参数

参数	参数类型	描述
error_msg	String	错误描述
error_code	String	错误码

请求示例

Get <https://{endpoint}/v4/projects/0792b7d26cb74bd799cbae80063dc74a/statuses>

响应示例

状态码： 200

OK

```
{
  "total": 1,
  "issue_statuses": [ {
    "id": "123cefde62004de2b62b5e8dd2c2b3af",
    "status_id": 1,
    "name": "新建",
    "tracker_ids": [ 0 ],
    "status_attribute": {
      "id": 1,
      "name": "string"
    }
  }
]
```

SDK 代码示例

SDK代码示例如下。

Java

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.projectman.v4.region.ProjectManRegion;
import com.huaweicloud.sdk.projectman.v4.*;
import com.huaweicloud.sdk.projectman.v4.model.*;

public class ListScrumProjectStatusesSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
        // security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
        // environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
        // this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        ProjectManClient client = ProjectManClient.newBuilder()
            .withCredential(auth)
            .withRegion(ProjectManRegion.valueOf("<YOUR REGION>"))
            .build();
        ListScrumProjectStatusesRequest request = new ListScrumProjectStatusesRequest();
        request.withOffset(<offset>);
        request.withLimit(<limit>);
        request.withTrackerId(<tracker_id>);
        try {
            ListScrumProjectStatusesResponse response = client.listScrumProjectStatuses(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

Python

```
# coding: utf-8

from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkprojectman.v4.region.projectman_region import ProjectManRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkprojectman.v4 import *

if __name__ == "__main__":
```

```
# The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
# In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
ak = __import__('os').getenv("CLOUD_SDK_AK")
sk = __import__('os').getenv("CLOUD_SDK_SK")

credentials = BasicCredentials(ak, sk) \

client = ProjectManClient.new_builder() \
    .with_credentials(credentials) \
    .with_region(ProjectManRegion.value_of("<YOUR REGION>")) \
    .build()

try:
    request = ListScrumProjectStatusesRequest()
    request.offset = <offset>
    request.limit = <limit>
    request.tracker_id = <tracker_id>
    response = client.list_scrum_project_statuses(request)
    print(response)
except exceptions.ClientRequestException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```

Go

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    projectman "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/projectman/v4/region"
)

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := projectman.NewProjectManClient(
        projectman.ProjectManClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListScrumProjectStatusesRequest{}
    offsetRequest := int32(<offset>)
    request.Offset = &offsetRequest
    limitRequest := int32(<limit>)
    request.Limit = &limitRequest
    trackerIdRequest := int32(<tracker_id>)
    request.TrackerId = &trackerIdRequest
    response, err := client.ListScrumProjectStatuses(request)
    if err == nil {
```

```
    fmt.Printf("%+v\n", response)
  } else {
    fmt.Println(err)
  }
}
```

更多

更多编程语言的SDK代码示例，请参见[API Explorer](#)的代码示例页签，可生成自动对应的SDK代码示例。

状态码

状态码	描述
200	OK
400	Bad Request
401	Unauthorized

错误码

请参见[错误码](#)。

5 历史 API (待下线)

5.1 项目成员角色待下线接口

CodeArts Req关于项目成员、项目成员角色增加、修改、删除、查看的功能即将下线，不推荐使用，接口列表如下：

```
[
  {
    "name": "成员加入项目审核开关",
    "method": "GET",
    "uri": "/projects/v1/member/{project_id}/invite-code"
  },
  {
    "name": "批量邀请其它企业用户加入项目",
    "method": "POST",
    "uri": "/projects/v3/projects/{project_id}/batch-invite"
  },
  {
    "name": "导出成员",
    "method": "GET",
    "uri": "/projects/v1/projects/{project_id}/member/export"
  },
  {
    "name": "批量查询当前用户在项目里的权限",
    "method": "POST",
    "uri": "/projects/v4/permission/projects/check"
  },
  {
    "name": "获取成员加入项目审核开关状态",
    "method": "GET",
    "uri": "/projects/v2/project/checkAuditSwitch"
  },
  {
    "name": "修改角色的权限",
    "method": "POST",
    "uri": "/projects/v2/permission/project_role"
  },
  {
    "name": "获取项目的成员列表",
    "method": "GET",
    "uri": "/projects/v2/member/list"
  },
  {
    "name": "主动退出项目",
    "method": "POST",
    "uri": "/projects/v1/member/quitProject"
  },
]
```

```
{
  "name": "处理审核列表中的用户",
  "method": "POST",
  "uri": "/projects/v2/member/audition"
},
{
  "name": "更新项目配置信息",
  "method": "POST",
  "uri": "/projects/v1/project/updateProjectConfig"
},
{
  "name": "批量导入用户",
  "method": "POST",
  "uri": "/projects/v4/users/batch-import"
},
{
  "name": "更新用户标签",
  "method": "POST",
  "uri": "/projects/v1/member/updateUserTag"
},
{
  "name": "调用console获取套餐详情",
  "method": "GET",
  "uri": "/projects/v4/{tenant_id}/package_detail"
},
{
  "name": "获取项目的成员列表",
  "method": "GET",
  "uri": "/projects/v2/member/{project_id}/all"
},
{
  "name": "管理项目成员列表",
  "method": "POST",
  "uri": "/projects/v4/members/management"
},
{
  "name": "角色变更申请列表",
  "method": "GET",
  "uri": "/projects/v4/roles/{project_id}/applications"
},
{
  "name": "申请角色变更",
  "method": "POST",
  "uri": "/projects/v4/roles/application"
},
{
  "name": "角色变更审批拒绝",
  "method": "POST",
  "uri": "/projects/v4/roles/{project_id}/application/rejection"
},
{
  "name": "角色变更审批同意",
  "method": "POST",
  "uri": "/projects/v4/roles/{project_id}/application/approval"
},
{
  "name": "批量查询成员信息",
  "method": "GET",
  "uri": "/projects/v4/projects/{project_id}/member/count_info"
},
{
  "name": "申请加入项目",
  "method": "POST",
  "uri": "/projects/v2/member/applyJoinMember"
},
{
  "name": "获取可以添加到项目的用户列表",
  "method": "GET",
  "uri": "/projects/v2/project/{project_id}/domain_users"
```

```
},
{
  "name": "查询项目下角色",
  "method": "GET",
  "uri": "/projects/v2/role/project_roles"
},
{
  "name": "批量删除成员",
  "method": "POST",
  "uri": "/projects/v1/member/deleteMembers"
},
{
  "name": "检查成员是否在项目中",
  "method": "GET",
  "uri": "/projects/v1/member/isMemberExistInProject"
},
{
  "name": "重新预约精简项目转换xboard项目",
  "method": "PUT",
  "uri": "/projects/v2/project/conversion/xboard"
},
{
  "name": "邀请其它企业用户加入项目",
  "method": "POST",
  "uri": "/projects/v2/member/invitation"
},
{
  "name": "解放号成员",
  "method": "POST",
  "uri": "/projects/v1/liberate/addMember"
},
{
  "name": "过滤条件列表",
  "method": "GET",
  "uri": "/projects/v4/members/management/conditions"
},
{
  "name": "获取通用设置->权限设置页面",
  "method": "GET",
  "uri": "/projects/v2/permission/common_permission_setting"
},
{
  "name": "更新昵称和性别",
  "method": "POST",
  "uri": "/projects/v1/user/updateCurrentUserInfo"
},
{
  "name": "修改是否可以创建项目的权限",
  "method": "POST",
  "uri": "/projects/v4/users/create-project-status"
},
{
  "name": "同一用户项目名称是否重复",
  "method": "GET",
  "uri": "/projects/v1/project/checkNameAndUser"
},
{
  "name": "根据项目id查询项目成员列表",
  "method": "GET",
  "uri": "/projects/v1/member/listWithUser"
},
{
  "name": "更新成员加入项目审核开关",
  "method": "POST",
  "uri": "/projects/v2/project/updateAuditSwitch"
},
{
  "name": "解放号获取签名",
  "method": "GET",
```

```
"uri": "/projects/v1/jf/sign"
},
{
  "name": "解放号检查",
  "method": "GET",
  "uri": "/projects/v1/jf/sign/check"
},
{
  "name": "取消精简项目转换xboard项目",
  "method": "POST",
  "uri": "/projects/v2/project/conversion/xboard"
},
{
  "name": "查询当前项目创建者租户是否授权domainName的租户",
  "method": "GET",
  "uri": "/projects/v2/member/authorization"
},
{
  "name": "管理项目成员列表",
  "method": "GET",
  "uri": "/projects/v4/members/management"
},
{
  "name": "更新昵称和性别",
  "method": "POST",
  "uri": "/projects/v1/user/updateUserNickName"
},
{
  "name": "获取审核中成员列表",
  "method": "GET",
  "uri": "/projects/v2/member/queryDomainAuditing"
},
{
  "name": "ProjectService",
  "method": "PUT",
  "uri": "/projects/v2/role/project_roles"
},
{
  "name": "修改自定义角色名字",
  "method": "POST",
  "uri": "/projects/v4/members/join"
},
{
  "name": "获取团队成员",
  "method": "GET",
  "uri": "/projects/v1/user/query_team_mates"
},
{
  "name": "获取项目设置->权限设置页面",
  "method": "GET",
  "uri": "/projects/v2/permission/service_permission_setting"
},
{
  "name": "查询当前用户角色",
  "method": "GET",
  "uri": "/projects/v1/user/getCurrentUserRole"
},
{
  "name": "项目类型初始化",
  "method": "POST",
  "uri": "/projects/v3/project/{project_id}/type_init/{type}"
},
{
  "name": "获取指定用户在项目里的角色和权限",
  "method": "GET",
  "uri": "/projects/v2/permission/role/{projectUUid}"
},
{
  "name": "批量更改成员角色",
```



```
"method": "POST",
"uri": "/projects/v1/member/updateRole"
},
{
"name": "批量删除成员",
"method": "POST",
"uri": "/projects/v1/member/batchDeleteMember"
},
{
"name": "更新项目",
"method": "POST",
"uri": "/projects/v1/project/update"
},
{
"name": "ProjectService",
"method": "GET",
"uri": "/projects/v2/member/same_projects"
},
{
"name": "和当前用户在同一个项目的所有成员",
"method": "GET",
"uri": "/projects/v1/user/get_team_mates_count"
},
{
"name": "批量更改成员角色",
"method": "POST",
"uri": "/projects/v1/member/batchUpdateMember"
},
{
"name": "批量添加成员",
"method": "POST",
"uri": "/projects/v1/member/importMembers"
},
{
"name": "批量添加成员",
"method": "POST",
"uri": "/projects/v1/member/addMembers"
},
{
"name": "管理项目成员",
"method": "POST",
"uri": "/projects/v4/members/management/batch-delete"
},
{
"name": "获取当前用户作为创建者、项目经理或者测试经理参与的项目列表",
"method": "GET",
"uri": "/projects/v2/project/project_type"
},
{
"name": "查询工作项处理人列表",
"method": "GET",
"uri": "/projects/v1/member/listIncludeRemovedWithUser"
},
{
"name": "删除自定义角色",
"method": "DELETE",
"uri": "/projects/v2/role/project_roles"
},
{
"name": "查询用户是否有创建项目的权限",
"method": "GET",
"uri": "/projects/v1/user/query_grant_create_project"
},
{
"name": "查询租户下计费的用户数",
"method": "GET",
"uri": "/projects/v4/usage/member_count"
},
{
}
```

```
    "name": "根据项目id查询项目成员列表",  
    "method": "GET",  
    "uri": "/projects/v1/member/queryMembers"  
  },  
  {  
    "name": "管理者看板查询项目列表",  
    "method": "POST",  
    "uri": "/projects/v1/project/queryManagerRelatedProject"  
  },  
  {  
    "name": "查询创建项目权限列表",  
    "method": "GET",  
    "uri": "/projects/v4/users/create-project-status"  
  },  
  {  
    "name": "取消精简项目转换xboard项目",  
    "method": "POST",  
    "uri": "/projects/v2/project/conversion/xboard/cancel"  
  },  
  {  
    "name": "新增自定义角色",  
    "method": "POST",  
    "uri": "/projects/v2/role/project_roles"  
  },  
  {  
    "name": "添加成员->其它企业用户->用户名校验",  
    "method": "GET",  
    "uri": "/projects/v1/member/checkTenantNameAndUserName"  
  },  
  {  
    "name": "创建用户标签",  
    "method": "POST",  
    "uri": "/projects/v1/member/createUserTag"  
  }  
]
```

6 应用示例

6.1 示例 1：获取需求管理中项目列表

场景描述

本章以用户test001为例，指导您通过API查询当前用户的项目列表。
API的调用方法请参见[如何调用API](#)。

约束限制

test001用户已经在需求管理中存在已自己创建的项目或参与的项目。

涉及接口

通过用户token查询当前用户项目信息。

URI: GET **/v4/projects**

API文档详情请参见：查询项目列表

API Explorer在线调试请参见：查询项目列表

操作步骤

- 请求示例
GET https://projectman-ext.cn-north-4.myhuaweicloud.com/v4/projects?
project_type=scrum&search=ProjectTestCase&offset=0&limit=1
- 响应示例

```
{
  "projects": [
    {
      "project_num_id": 3844012,
      "project_id": "ec147b681fb64e019a6bcd73ebdf3597",
      "project_name": "ProjectTestCaseYRXI",
      "description": "fdas",
      "created_time": 1600417972000,
      "updated_time": 1600417972000,
      "project_type": "scrum",
      "creator": {
        "user_num_id": 4090,
```

```
"user_id": "665b542eb4c14b53a15ead449f28b0ba",
"user_name": "test001",
"domain_id": "70d11a2d7275425689347ec104af2ad8",
"domain_name": "projectman_l00382124_01",
"nick_name": "英国队长"
  }
}
],
"total": 48
}
```

6.2 示例 2：查询指定项目下的迭代列表

场景描述

本章以test001为例，指导您通过API查询指定项目下的迭代列表。
API的调用方法请参见[如何调用API](#)。

约束限制

test001用户已经在需求管理中存在已自己创建的项目。

涉及接口

通过用户token查询当前用户指定项目下的迭代列表接口信息

URI: GET **/v4/projects/{project_id}/iterations**

API文档详情请参见：获取指定项目的迭代列表

API Explorer在线调试请参见：获取指定项目的迭代列表

操作步骤

- 请求示例
GET <https://projectman-ext.cn-north-4.myhuaweicloud.com/v4/projects/3cea7d4d77134eee8251825918a8bd7a/iterations>

- 响应示例

```
{
  "iterations": [
    {
      "id": 707961817,
      "name": "迭代1",
      "description": "",
      "begin_time": "2020/06/07",
      "end_time": "2020/07/06"
    },
    {
      "id": 707961818,
      "name": "迭代2",
      "description": "",
      "begin_time": "2020/07/07",
      "end_time": "2020/08/06"
    },
    {
      "id": 707961819,
      "name": "迭代3",
      "description": ""
    }
  ]
}
```

```
"begin_time": "2020/08/07",  
"end_time": "2020/09/06"  
}  
],  
"total": 3  
}
```

6.3 示例 3：查询用户工作项的评论

场景描述

test001用户已经在需求管理中存在已自己创建的项目或参与的项目，项目中有已经创建好的工作项。

API的调用方法请参见[如何调用API](#)。

约束限制

工作项存在。

涉及接口

先通过用户token查询指定项目的工作项列表，再通过项目id、工作项id查询工作项评论，步骤如下：

1. [步骤1：查询用户的工作项列表](#)
2. [步骤2：查询工作项的评论](#)

步骤 1：查询用户的工作项列表

- 接口信息
URI: Post /v4/projects/{project_id}/issues
API文档详情请参见：[高级查询工作项](#)
API Explorer在线调试请参见：[高级查询工作项](#)
- 请求示例
Post <https://projectman-ext.cn-north-4.myhuaweicloud.com/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues>
- 请求体：

```
{  
  "status_ids": [1],  
  "limit": 1  
}
```
- 响应示例

```
{  
  "issues": [  
    {  
      "id": 1692040,  
      "name": "测试",  
      "done_ratio": 0,  
      "tracker": {  
        "id": 2,
```

```
"name": "Task"
  },
  "priority": {
    "id": 2,
    "name": "中"
  },
  "severity": {
    "id": 12,
    "name": "一般"
  },
  "status": {
    "id": 1,
    "name": "新建"
  },
  "updated_time": "2020-09-17 10:18:31",
  "begin_time": "1970-01-01",
  "end_time": "1970-01-01",
  "project": {
    "project_num_id": 3694732,
    "project_id": "526cefde62004de2b62b5e8dd2c2b3af",
    "project_name": "sdfsdfsf"
  },
  "parent_issue": {
    "id": 1684959,
    "name": "testDemo阿达萨达"
  },
  "module": {
    "id": null,
    "name": null
  },
  "domain": {
    "id": null,
    "name": null
  },
  "iteration": {
    "id": null,
    "name": null
  },
  "expected_work_hours": 0.0,
  "actual_work_hours": 0.0,
  "created_time": "2020-09-17 10:18:31",
  "custom_fields": [],
  "assigned_cc_user": null,
  "assigned_user": {
    "id": 4091,
    "name": "test001",
    "nick_name": "pcedgeXXXXXXX"
  },
  "creator": {
    "id": 4091,
    "name": "test001",
    "nick_name": "pcedgeXXXXXXX"
  },
  "developer": {
    "id": 0,
    "name": null,
    "nick_name": null
  }
},
"total": 24
}
```

步骤 2: 查询工作项的评论

- 接口信息
URI: GET [/v4/projects/{project_id}/issues/{issue_id}/comments](#)
API文档详情请参见: [获取指定工作项的评论列表](#)

API Explorer在线调试请参见：[获取指定工作项的评论列表](#)

- 请求示例

GET <https://projectman-ext.cn-north-4.myhuaweicloud.com/v4/projects/526cefde62004de2b62b5e8dd2c2b3af/issues/1692040/comments>

- 响应示例

```
{
  "comments": [
    {
      "id": 3235092,
      "created_time": "2020-08-26",
      "user": {
        "user_num_id": 4091,
        "user_name": "test001",
        "nick_name": "pcedgeXXXXXX"
      },
      "comment": "<p>这是一个测试样例</p>"
    }
  ],
  "total": 1
}
```

7 附录

7.1 状态码

状态码如表7-1所示。

表 7-1 状态码

状态码	编码	错误码说明
100	Continue	继续请求。 这个临时响应用来通知客户端，它的部分请求已经被服务器接收，且仍未被拒绝。
101	Switching Protocols	切换协议。只能切换到更高级的协议。 例如，切换到HTTP的新版本协议。
201	Created	创建类的请求完全成功。
202	Accepted	已经接受请求，但未处理完成。
203	Non-Authoritative Information	非授权信息，请求成功。
204	NoContent	请求完全成功，同时HTTP响应不包含响应体。 在响应OPTIONS方法的HTTP请求时返回此状态码。
205	Reset Content	重置内容，服务器处理成功。
206	Partial Content	服务器成功处理了部分GET请求。
300	Multiple Choices	多种选择。请求的资源可包括多个位置，相应可返回一个资源特征与地址的列表用于用户终端（例如：浏览器）选择。
301	Moved Permanently	永久移动，请求的资源已被永久的移动到新的URI，返回信息会包括新的URI。

状态码	编码	错误码说明
302	Found	资源被临时移动。
303	See Other	查看其它地址。 使用GET和POST请求查看。
304	Not Modified	所请求的资源未修改，服务器返回此状态码时，不会返回任何资源。
305	Use Proxy	所请求的资源必须通过代理访问。
306	Unused	已经被废弃的HTTP状态码。
400	BadRequest	非法请求。 建议直接修改该请求，不要重试该请求。
401	Unauthorized	在客户端提供认证信息后，返回该状态码，表明服务端指出客户端所提供的认证信息不正确或非法。
402	Payment Required	保留请求。
403	Forbidden	请求被拒绝访问。 返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
404	NotFound	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。
405	MethodNotAllowed	请求中带有该资源不支持的方法。 建议直接修改该请求，不要重试该请求。
406	Not Acceptable	服务器无法根据客户端请求的内容特性完成请求。
407	Proxy Authentication Required	请求要求代理的身份认证，与401类似，但请求者应当使用代理进行授权。
408	Request Time-out	服务器等候请求时发生超时。 客户端可以随时再次提交该请求而无需进行任何更改。
409	Conflict	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
410	Gone	客户端请求的资源已经不存在。 返回该状态码，表明请求的资源已被永久删除。
411	Length Required	服务器无法处理客户端发送的不带Content-Length的请求信息。

状态码	编码	错误码说明
412	Precondition Failed	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。
413	Request Entity Too Large	由于请求的实体过大，服务器无法处理，因此拒绝请求。为防止客户端的连续请求，服务器可能会关闭连接。如果只是服务器暂时无法处理，则会包含一个Retry-After的响应信息。
414	Request-URI Too Large	请求的URI过长（URI通常为网址），服务器无法处理。
415	Unsupported Media Type	服务器无法处理请求附带的媒体格式。
416	Requested range not satisfiable	客户端请求的范围无效。
417	Expectation Failed	服务器无法满足Expect的请求头信息。
422	Unprocessable Entity	请求格式正确，但是由于含有语义错误，无法响应。
429	TooManyRequests	表明请求超出了客户端访问频率的限制或者服务端接收到多于它能处理的请求。建议客户端读取相应的Retry-After首部，然后等待该首部指出的时间后再重试。
500	InternalServerError	表明服务端能被请求访问到，但是不能理解用户的请求。
501	Not Implemented	服务器不支持请求的功能，无法完成请求。
502	Bad Gateway	充当网关或代理的服务器，从远端服务器接收到了一个无效的请求。
503	ServiceUnavailable	被请求的服务无效。 建议直接修改该请求，不要重试该请求。
504	ServerTimeout	请求在给定的时间内无法完成。客户端仅在为请求指定超时（Timeout）参数时会得到该响应。
505	HTTP Version not supported	服务器不支持请求的HTTP协议的版本，无法完成处理。

7.2 错误码

调用接口出错后，将不会返回结果数据。调用方可根据每个接口对应的错误码来定位错误原因。

当调用出错时，如HTTP请求返回一个400或505的HTTP状态码。返回的消息体中是具体的错误代码及错误信息。

在调用方找不到错误原因时，可以联系华为云客服，并提供错误码，以便我们尽快帮您解决问题。

错误响应 Body 体格式说明

当接口调用出错时，会返回错误码及错误信息说明，错误响应的Body体格式如下所示。

```
{
  "error_msg": "The format of message is error",
  "error_code": "AS.0001"
}
```

其中，“error_code”表示错误码，“error_msg”表示错误描述信息。

错误码说明

当您调用API时，如果遇到“APIGW”开头的错误码，请参见[API网关错误码](#)进行处理。

更多服务错误码请参见[API错误中心](#)。

表 7-2 错误码

状态码	错误码	错误信息	描述	处理措施
400	PM.02150010	目标用户非项目空间成员	目标用户非项目空间成员	将需要的用户先添加到项目空间
400	PM.02150301	删除失败	删除失败	将是这个角色的用户先修改成其他的角色，再删除
400	PM.02150302	所设置权限不可变更	所设置权限不可变更	所设置权限不可变更
400	PM.02150303	角色不存在	角色不存在	核实请求参数
400	PM.02150304	角色名不可用，请设置新名称	角色名不可用，请设置新名称	请设置新名称
400	PM.02150305	角色名冲突	角色名冲突	请设置新名称
400	PM.02150404	项目空间不存在	项目空间不存在	项目空间不存在
400	PM.02151001	项目集不存在	项目集不存在	项目集不存在
400	PM.02152001	参数错误	参数错误	参数错误
400	PM.02152002	未定义错误	未定义错误	未定义错误

状态码	错误码	错误信息	描述	处理措施
400	PM.02152003	传入参数名非法	传入参数名非法	传入参数名非法
400	PM.02152004	项目集未归档	项目集未归档	项目集未归档
400	PM.02152005	项目不存在或项目已被删除	项目不存在或项目已被删除	项目不存在或项目已被删除
400	PM.02152006	该项目集还没有被关注	该项目集还没有被关注	该项目集还没有被关注
400	PM.02152007	已关注该项目集	已关注该项目集	已关注该项目集
400	PM.02152008	修改创建项目集权限的参数含有非法用户	修改创建项目集权限的参数含有非法用户	修改创建项目集权限的参数含有非法用户
400	PM.02152009	项目集已归档	项目集已归档	项目集已归档
400	PM.02152010	项目类型不被允许	项目类型不被允许	项目类型不被允许
400	PM.02152011	项目已被关联	项目已被关联	项目已被关联
400	PM.02153001	访问频率过高, 请稍后再试	访问频率过高, 请稍后再试	访问频率过高, 请稍后再试
400	PM.02175006	CN: 参数错误: {0}; EN: Invalid Parameter: {0}	参数错误	使用正确的请求参数重试
400	PM.02175101	CN: 子节点数量超过限制: {0}; Number of sub items exceeds [{0}]	子节点数量不能超过限制	删除其他子节点后再创建, 或者在其他父节点下创建
400	PM.02175103	CN: 无法修改未被接纳或已拒绝的IR; EN: Only accepted initial requirement can be edited	当前IR无法被修改 (只有当提出项目空间与归属项目空间相同, 或者在归属项目空间被接纳的IR才能被修改。修改IR或者在父IR下创建子项时可能发生此错误)	确认修改的IR的状态后重试

状态码	错误码	错误信息	描述	处理措施
400	PM.02175 104	CN: 所选IR无法操作; EN: Some of the initial requirements can not be edited	部分IR无法修改 (通常是因为部分 请求的IR不存在)	确认所有修改 的IR都存在后 重试
400	PM.02175 105	CN: 无效的父IR; EN: Invalid parent.	无效的父IR (此错 误发生于批量创建 子IR时, 父IR不一 致, 或者父IR的归 属项目空间与子IR 的提出项目空间不 一致, 或者指定的 父IR类型不正确)	使用正确的请 求参数重试
400	PM.02175 106	CN: 子IR必须提给其他 项目空间; EN: Can not create sub ir to current program.	子IR的提出项目空 间与归属项目空间 不能相同	将子IR提给其 他项目空间
400	PM.02175 111	CN: 当前IR下有子IR, 不能删除; EN: Child IR exist, can not delete.	不能删除具有子IR 的父IR	删除所有子IR 后再删除父IR
400	PM.02175 113	CN: 父IR不存在; EN: Parent IR not exists.	父IR不存在	在正确的父IR 下创建子IR
400	PM.02175 114	CN: 父IR不归属于当前 项目空间, 请到父IR 的归属项目空间进行 操作; EN: Parent IR not designated in current program, please perform in the destination program of this IR.	父IR不归属于当前 项目空间	到父IR的归属 项目空间中重 试
400	PM.02175 115	CN: 不能删除IR: 不具 有所有子工作项所在 项目的编辑权限; EN: Missing edit permission of sub items.	不能删除IR, 因为 不具有所有子工作 项所在项目的编辑 权限	申请子工作项 所在项目的编 辑权限后重试
400	PM.02175 200	CN: 缺少必填字段 [{}]; EN: Missing required field [{}]	缺少必填字段	添加必填字段 后重试

状态码	错误码	错误信息	描述	处理措施
400	PM.02175 201	CN: 字段 [{0}] 的值不正确{1}; EN: Field [{0}] has invalid value{1}	IR字段的值不正确（无具体原因是，通常是格式错误，比如时间/日期字段使用了非时间戳字符串）	使用正确的值进行请求
400	PM.02175 202	CN: 预计开始日期不能晚于预计结束日期; EN: Plan start date can not be later than plan due date.	预计开始日期不能晚于预计结束日期	使用正确的预计开始日期/预计结束日期参数
400	PM.10001 002	不合法的token	不合法的token	不合法的token
400	PM.10002 006	当前租户下名称已被使用	当前租户下名称已被使用	当前租户下名称已被使用
400	PM.10002 010	描述不超过500个字符	描述不超过500个字符	描述不超过500个字符
400	PM.10002 011	名称不规范	名称不规范	名称不规范
400	PM.10003 002	当前用户权限不足	当前用户权限不足	当前用户权限不足
400	PM.10003 003	邀请已过期	邀请已过期	邀请已过期
400	PM.10003 004	审核信息不存在	审核信息不存在	审核信息不存在
400	PM.10003 005	标签在使用中	标签在使用中	标签在使用中
400	PM.10003 006	添加成员最多不超过100个	添加成员最多不超过100个	添加成员最多不超过100个
400	PM.10003 007	最多可以添加100个项目	最多可以添加100个项目	最多可以添加100个项目
400	PM.10003 008	角色不被允许	角色不被允许	角色不被允许
400	PM.10003 009	创建者移交项目空间后可以退出项目	创建者移交项目空间后可以退出项目	创建者移交项目空间后可以退出项目
400	PM.10003 010	成员已不存在	成员已不存在	成员已不存在

状态码	错误码	错误信息	描述	处理措施
400	PM.10003011	租户下项目空间总数不能超过100个	租户下项目空间总数不能超过100个	租户下项目空间总数不能超过100个
400	PM.10003014	邀请不存在	邀请不存在	邀请不存在
400	PM.10003015	只能添加同租户下用户	只能添加同租户下用户	只能添加同租户下用户
400	PM.10003016	添加成员失败	添加成员失败	添加成员失败
401	PM.02150401	用户操作权限不足	用户操作权限不足	用户操作权限不足
403	PM.10003012	用户是项目空间成员，不需要申请	用户是项目空间成员，不需要申请	用户是项目空间成员，不需要申请
404	PM.02175005	CN: 当前请求资源不存在; EN: Requested resource not exist	查询的资源不存在	确认请求的资源是否存在（可能输入了错误的id，或资源已被删除）
405	PM.10003013	已存在加入项目空间的申请，等待管理员审批	已存在加入项目空间的申请，等待管理员审批	已存在加入项目空间的申请，等待管理员审批

7.3 获取项目 ID

调用 API 取项目 ID

项目ID还用通过调用[查询指定条件下的项目信息](#)API获取。

获取项目ID的接口为“GET https://{Endpoint}/v3/projects/”，其中“{Endpoint}”为IAM的终端节点，可以从[终端节点](#)获取。接口的认证鉴权请参见[认证鉴权](#)。

响应示例如下，其中“projects”下的“id”即为项目ID。

```
{
  "projects": [
    {
      "domain_id": "65382450e8f64ac0870cd180d14e684b",
      "is_domain": false,
      "parent_id": "65382450e8f64ac0870cd180d14e684b",
      "name": "cn-north-4",

      "description": "",
      "links": {
```

```
    "next": null,
    "previous": null,
    "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
  },
  "id": "a4a5d4098fb4474fa22cd05f897d6b99",
  "enabled": true
}
],
"links": {
  "next": null,
  "previous": null,
  "self": "https://www.example.com/v3/projects"
}
}
```

从控制台获取项目 ID

在调用接口的时候，部分URL中需要填入项目编号，所以需要获取到项目编号。项目编号获取步骤如下：

1. 登录管理控制台。
2. 鼠标悬停在右上角的用户名，在下拉列表中单击“我的凭证”。进入“我的凭证 > API凭证”页面，在项目列表中查看项目ID。

图 7-1 查看项目 ID



说明

多项目时，展开“所属区域”，从“项目ID”列获取子项目ID。

7.4 获取账号 ID

在调用接口的时候，部分URL中需要填入账号ID，所以需要先在管理控制台上获取到账号ID。账号ID获取步骤如下：

1. 登录管理控制台。
2. 鼠标悬停在右上角的用户名，在下拉列表中单击“我的凭证”。进入“我的凭证 > API凭证”页面，即可查看账号ID。

8 文档修订记录

文档修订记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本的更新内容。

修订记录	发布日期
第二十五次正式发布。本次变更说明如下：增加了SDK代码示例。	2024-03-08
第二十四次正式发布。本次变更说明如下： <ul style="list-style-type: none">增加了添加指定工作项工时。增加了查询项目下的工时类型。增加了获取指定工作项停留时间。增加了高级查询我的待办工作项。增加了查询项目模板。	2023-06-08
第二十三次正式发布。本次变更说明如下： <ul style="list-style-type: none">issueUser对象中增了两个字段：user_id和user_num_id。	2022-12-06
第二十二次正式发布。本次变更说明如下： <ul style="list-style-type: none">增加了下载工作项附件。增加了删除附件。增加了查询当前工作项已经关联的代码提交记录 / 分支创建记录。	2022-11-04
第二十一次正式发布。本次变更说明如下： <ul style="list-style-type: none">增加响应示例和请求示例。	2022-10-28
第二十次正式发布。本次变更说明如下： <ul style="list-style-type: none">增加了下载图片。增加了上传工作项附件。增加了Scrum项目的领域。	2022-09-16

修订记录	发布日期
第十九次正式发布。本次变更说明如下： <ul style="list-style-type: none">• 修改了API概览。• 修改了终端节点。• NewCustomField中增加了字段“field_name”。	2022-08-23
第十八次正式发布。本次变更说明如下： <ul style="list-style-type: none">• 修改了API概览。• 增加了Scrum项目的模块。	2022-06-30
第十七次正式发布。本次变更说明如下： <ul style="list-style-type: none">• 增加了查询项目的工作项。• 增加了查询项目下所有工作项的历史记录。• 增加了查询Scrum项目的工作项流转配置。• 增加了查询当前工作项已经关联的工作项。• 增加了查询当前工作项已经关联的关联Wiki。• 增加了查询当前工作项已经关联的代码提交记录 / 分支创建记录。• 增加了查询关联用例。• 增加了查询Scrum工作项自定义字段。• 增加了查询看板项目下工作项的状态历史记录。• 增加了查询看板项目下的工作项。• 增加了查询看板项目的工作项流转配置。• 修改了创建工作项。	2022-06-06
第十六次正式发布。本次变更说明如下： <ul style="list-style-type: none">• 增加了查看迭代历史记录。• 增加了更新子用户昵称。• 修改了高级查询工作项。• 修改了获取指定项目的迭代列表。	2022-02-28
第十五次正式发布。本次变更说明如下： <ul style="list-style-type: none">• 修改了高级查询工作项。• 修改查询缺陷密度。	2021-11-26
第十四次正式发布。本次变更说明如下： <ul style="list-style-type: none">• 增加了细粒度权限用户创建工作项。• 修改了错误码。	2021-11-10
第十三次正式发布。本次变更说明如下： 修改了API字段信息。 <ul style="list-style-type: none">• 迭代详情和迭代列表添加了返回字段status。• 获取工作项的历史记录添加了id和name。	2021-09-28

修订记录	发布日期
第十二次正式发布。本次变更说明如下： <ul style="list-style-type: none">修改了错误码。	2021-06-11
第十一次正式发布。本次变更说明如下： <ul style="list-style-type: none">修改了错误码。	2021-05-12
第十次正式发布。本次变更说明如下： <ul style="list-style-type: none">修改了错误码。	2021-05-08
第九次正式发布。本次变更说明如下： <ul style="list-style-type: none">修改了接口字段信息。	2021-04-22
第八次正式发布。本次变更说明如下： <ul style="list-style-type: none">删除约束与限制。	2021-03-26
第七次正式发布。本次变更说明如下： <ul style="list-style-type: none">修改API概览。	2021-03-10
第六次正式发布。本次变更说明如下： <ul style="list-style-type: none">修改应用示例。	2021-02-05
第五次正式发布。本次变更说明如下： <ul style="list-style-type: none">增加应用示例。	2020-08-31
第四次正式发布。本次变更说明如下： <ul style="list-style-type: none">修改“4.1 API”。	2020-08-18
第三次正式发布。本次变更说明如下： <ul style="list-style-type: none">修改基本概念。修改构造请求。修改认证鉴权。	2020-08-06
第二次正式发布。本次变更说明如下： 修改 终端节点 。	2020-07-01
第一次正式发布。	2020-03-30